RL-TR-97-22
Final Technical Report
June 1997

ALGORITHMIC SOLUTIONS FOR COUPLED THERMAL AND ELECTROMAGNETIC ANALYSIS

Sabbagh Associates, Inc.

Harold A. Sabbagh, Lai Won Woo, and Xun Yang

Rome Laboratory
Air Force Materiel Command
Rome, New York

Although this report references a limited document (*), listed on page 65, no limited information has been extracted.

This report has been reviewed by the Rome Laboratory Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RL-TR-97-22 has been reviewed and is approved for publication.

APPROVED: Dale W. Richards

DALE W. RICHARDS
Project Engineer

FOR THE COMMANDER: John J. Bart

JOHN J. BART, Chief Scientist
Electromagnetics & Reliability Directorate

If your address has changed or if you wish to be removed from the Rome Laboratory mailing list, or if the addressee is no longer employed by your organization, please notify RL/ERSR, 525 Brooks Road, Rome, NY 13441-4505. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June 1997	Final Aug 95 - Sep 96

4. TITLE AND SUBTITLE

ALGORITHMIC SOLUTIONS FOR COUPLED THERMAL AND ELECTROMAGNETIC ANALYSIS

5. FUNDING NUMBERS

C - F30602-95-C-0249
PE - 62702F
PR - 2338
TA - 02
WU - PP

6. AUTHOR(S)

Harold A. Sabbagh, Lai Won Woo, and Xun Yang

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Sabbagh Associates, Inc.
4635 Morningside Drive
Bloomington, IN 47408

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Rome Laboratory/ERSR
525 Brooks Road
Rome, NY 13441-4505

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

RL-TR-97-22

11. SUPPLEMENTARY NOTES

Rome Laboratory Project Engineer: Dale W. Richards/ERSR/(315) 330-3014

12a. DISTRIBUTION AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This effort investigated the strengths and weaknesses of current attempts at integrating data-representation and analysis techniques for electrical, electromagnetic, thermal and structural analysis of microelectronic components, principally microwave modules. Potential contributions and risks of various integration approaches were assessed. Constraints and opportunities associated with software implementation of this technology were also investigated. Guidelines, based on analytical and empirical studies, for efficiently solving coupled EM and TM problems were developed. Initial work focused on defining a suite of canonical models. Once these models were established, analytical studies were performed to identify constraints associated with integrated analyses of these models. Selected parametric values were varied to determine their effects on the analysis and to assess the sensitivity of the models. Of particular concern was the potential for the analysis to converge. A sequence-accelerating algorithm was applied to speed the convergence process. Empirical studies were performed to assess the conditions under which it was more appropriate to use two separate gridding structures rather than a single grid for both electrical and thermal analysis. Issues of adaptive meshing and extrapolation of solutions were also investigated.

14. SUBJECT TERMS

Coupled Solution, Computational Electromagnetics, Thermo-mechanical, Finite Element Analysis, FEA, Modeling, Gridding, Microwave

15. NUMBER OF PAGES

80

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Contents

List of Tables

List of Figures

0 EXECUTIVE SUMMARY

The design of microwave components is driven by electromagnetic considerations, but these devices always fail because of thermomechanical weaknesses. Therefore, it is necessary to apply concurrent engineering principles early in the design phase to ensure that all aspects of the design are being met. The major issues addressed in this report deal with current practices for integrating data-representation and analysis techniques for coupled computational electromagnetic (CEM) and thermomechanical (TM) problems, solving canonical coupled TM and CEM problems, and related matters, such as mesh refinement.

Current Practices: In a typical example of software for solving coupled problems, as described in Chapter 2, a supervisor, called FLYSTRYD, is required to control the activities of the various analysis codes.

Instead of just transferring the interesting data from the output files of one application software to the input files of another, the supervisor carries out that transfer between different meshes and geometries. A consequence of this is that FLYSTRYD must handle two data-bases at the same time. Those data-bases contain two different models of the same physical problem—for instance, electromagnetic and mechanical models. This allows the supervisor to interpolate important physical quantities on different meshes. Interpolation is performed, as usual, by way of the shape functions of the finite elements. The difficulty arises because of the wide variety of types of elements of the different analysis codes.

When the geometries of the various physical models differ, one has to use projection or extrapolation techniques to provide the correct data. Projection is used when a physical quantity is defined on a source region which does not exist in the target model. On the other hand, extrapolation is necessary to define a quantity in a region that did not exist in the source domain.

Analysis of Canonical Problems: The two principal canonical problems chosen for this study involve thermo-electromagnetic interactions in a parallel plate capacitor, whose plates are separated by a material whose electrical conductivity varies with temperature and electric field. This makes the electrical problem nonlinear, but the thermal problem remains linear.

We developed an algorithm to solve coupled CEM and TM problems using the finite-element code, TOPAZ, and found it to be stable under a wide range of electical and thermal parameters. We then introduced the theta-algorithm, which is a sequence-accelerating algorithm, and found that the limit of the slowly converging sequence of TOPAZ iterates can be found quickly and efficiently.

In a final study, we attempted to force a condition that might require different grids to be used for the electrical and thermal problems, without changing the geometry or boundary conditions. That such a condition might exist rests with the nonlinear electrical conductivity profile of the material. We found that when T is about 60° C and E is about 1170 V/m, the largest change in temperature between two adjacent nodes is 2.7%, and the largest change of electric field is 0.7%. Thus, there is no need to change the grid. When T is about 120° C and E is about a few hundred V/m, however, the largest change of temperature between two adjacent nodes is 3.4%, and the largest change of electric field is about 20%. Now, it may be necessary to use different grids to solve the electrical and thermal problems.

Related Matters: We introduce the concepts of *a priori* and *a posteriori* error indicators, and explain their use in adaptive mesh generation. An *a priori* error estimate is one that is theoretically known before a computation is made. The coefficients in the expansion for the error estimate, however, may not be known. Such error estimates may not be precise enough to be used in practice without making some computations, i.e., performing a numerical experiment. The resulting error

estimate then becomes *a posteriori*, that is, after the experiment is performed.

Furthermore, we define an optimal mesh: A mesh is said to be optimal when the measure of error in the solution is equal for each element in the mesh. Those measures of error that are commonly used are the energy norm, relative percentage energy norm error, and local and global effectivity indices. The effectivity index, $\Theta = \|e\|/\|e\|_{ex}$, where $\|e\|$ and $\|e\|_{ex}$ are computed and exact errors, respectively, in the energy norm.

The mesh refinement algorithm can be made efficient through the use of the *hp*-adaptive analysis technique. In this technique, the mesh width, h, or the order of the approximating polynomial, p, are adapted to the problem at hand in such a way that an optimal mesh is generated.

Conclusion: 'Grid-reusability,' that is the use or reuse of grid-related data depends upon problem conditions that are as diverse as geometry, the physical model, and materials properties. Nonlinearities that arise when a physical parameter depends upon the value of the field variable being sought can thoroughly complicate gridding requirements to such an extent that *a priori* guidelines are not available. The grid and its associated data will vary as the solution is computed iteratively.

The question of grid-reusability has surfaced only rather recently, as designers begin to address the question of concurrent engineering in a rigorous, self-consistent manner. We have shown a few examples in which researchers are addressing this aspect of CAE software development, but we are unaware of this problem being dealt with in commercial software.

The grand challenge of the immediate future is to extend the research reported herein to the problem of doing adaptive meshing on two or more grids for the purpose of optimally solving coupled problems with nonlinearities. What role will *hp*-adaptive analysis play in meeting this challenge?

1 INTRODUCTION

The design of microwave components is driven by electromagnetic considerations, but these devices always fail because of thermomechanical weaknesses. Therefore, it is necessary to apply concurrent engineering principles early in the design phase to ensure that all aspects of the design are being met. This includes, of course, the underlying electromagnetic requirements, as well as thermo-mechanical considerations. This is especially true as components are being designed with an ever increasing density of active and inactive, heat generating, elements. From the perspective of the designer of microwave components, the problem now involves a number of different physical models which come into play: electromagnetics, thermomechanics, and perhaps fluid flow. Now the design problem involves computational electromagnetics (CEM), computational mechanics, and perhaps computational fluid dynamics (CFD). Each of these areas requires its own data base and computational algorithm. From the perspective of the software designer of a computer-aided engineering (CAE) package, the problem becomes one of efficiently communicating between the algorithms and data bases that each of the physical models requires.

The research effort reported herein addressed the rather broad area of algorithms and software needs for solving coupled problems by focusing on the following issues:

1. Investigate current practices for integrating data-representation and analysis techniques for various microelectronics-related engineering disciplines, such as computational electromagnetics and thermomechanics.

2. Establish a set of canonical test problems whose purpose is to determine the goodness qualities of solutions involving the coupling of electromagnetic and thermo-mechanical analyses, that are typically encountered in the design of T/R modules. Conduct an empirical (computational) study using these models in order to learn as much as possible about the intricacies of solving such coupled problems.

3. Investigate related matters, such as adaptive meshing, error analysis, and extrapolation of solutions.

Chapter 2 of this report deals with current practices for integrating data-representation and analysis techniques for coupled thermomechanical and electromagnetic problems. In a typical example described in Chapter 2, a supervisor, called FLYSTRYD, is required to control the activities of the various analysis codes.

Instead of just transferring the interesting data from the output files of one application software to the input files of another, the supervisor carries out that transfer between different meshes and geometries. A consequence of this is that FLYSTRYD must handle two data-bases at the same time. Those data-bases contain two different models of the same physical problem—for instance, electromagnetic and mechanical models. This allows the supervisor to interpolate important physical quantities on different meshes. Interpolation is performed, as usual, by way of the shape functions of the finite elements. The difficulty arises because of the wide variety of types of elements of the different analysis codes.

When the geometries of the various physical models differ, one has to use projection or extrapolation techniques to provide the correct data. Projection is used when a physical quantity is defined on a source region which does not exist in the target model. On the other hand, extrapolation is necessary to define a quantity in a region that did not exist in the source domain.

The second item is the central element of this project, and is discussed in Chapter 3. The two principal canonical problems chosen for this study involve thermo-electromagnetic interactions in

a parallel plate capacitor, whose plates are separated by a material whose electrical conductivity varies with temperature and electric field. In the initial numerical experiments, the material is taken to be Barium-Titanate, but the conductivity is then changed later in order to determine its effects on the solution of the problem.

The geometry of the problems is rather simple, but the mesh requirements may still become complicated by the physics involved. For example, the thermal convection coefficient can play a significant role in the mesh requirements, even though the geometry is quite regular. Because the electrical conductivity is a function of the electric field, the problem is nonlinear in the electrical variables, and because the same conductivity is also a function of temperature, the electrical and thermal problems are coupled. In these problems, the thermal conductivity is assumed to be a constant, regardless of the temperature and electric field. Hence, the thermal problem is linear.

The nonlinearities can introduce some significant effects on the grid requirements for the thermal and electromagnetic problems. The electrical conductivity of Barium Titanate exhibits distinctly different behavior in two temperature ranges. In the lower range (between 0°C and 100°C), the conductivity increases slightly with temperature, which is typical of a semiconductor, while in the upper range (above 100°C), the conductivity decreases, which is reminiscent of a metal.

Our results are summarized as:

(a) In the range $0°C < T < 100°C$, for example, when T is about 60 °C and E is about 1170 V/m, the largest change of temperature between two adjacent nodes is 2.7% and the largest change of electrical field is about 0.7%, and there is no need to change the grid.

(b) In the range $100°C < T < 140°C$, for example, when T is about 120 °C and E is about a few hundred V/m, the largest change of temperature between two adjacent nodes is 3.4%, and the largest change of electrical field is about 20%. In this case, it may be necessary to use two different grids to solve the electrical problem and the thermal problem.

When the temperature is above 100°C, a slight change of temperature will cause a large change of electric field, and this is depends solely on the non-linearity of the material. One must consider several factors in order to get the temperature above 100°C for this particular material, but the thermal load $(\sigma(E,T) * E^2)$ is the most important one. The ambient temperature, initial temperature, and convection coefficient also play key roles in this problem.

For instance:

(a) When the thermal load is about 3.0×10^6 to 4.0×10^6, we need to consider using different grids. It doesn't matter what the ambient temperature is as long as the convection coefficient is small enough to keep the temperature in the range between 100°C and 140°C. For this particular type of material, if we rescale the electrical conductivity σ by a factor of 1.67, the thermal load will then be in the range of 3.0×10^6 to 4.0×10^6, and the results shown in the second table are obtained (a relatively big change in E-field and small change in T).

(b) If the ambient temperature is in the region of 100°C to 140°C, and if the convection coefficient is large enough to keep the temperature within the body close to the ambient temperature, then it may be necessary to use different grids for the T and E calculation. Ambient temperatures, however, are normally around 20°C, so that this case is rare.

(c) In conclusion, therefore, the question of whether or not to change the grid for the E and T calculation depends upon the non-linearity of the material for a problem with a simple geometry, such as Canonical Problem No. 2. For Canonical Problem No. 3, however, the situation is very different and further studies should be done.

We introduce the θ-algorithm in Chapter 3, and demonstrate its ability to accelerate the convergence of the iterations between the thermal and electrical aspects of the coupled problem.

The first part of Chapter 4 consists of a summary and literature search that deal with a number of things, such as the display of scalars and vectors, to a review of hardware and software, to the very important subject of mesh generation. These topics are of fundamental importance in relating computational electromagnetics to computer-aided engineering and design. The second part of the chapter deals with the subject of extrapolation of solutions to reduce the effects of truncation error caused by meshing. We introduce the concepts of *a priori* and *a posteriori* error indicators, and explain their use in adaptive mesh generation. An *a priori* error estimate is one that is theoretically known before a computation is made. The coefficients in the expansion for the error estimate, however, may not be known. Such error estimates may not be precise enough to be used in practice without making some computations, i.e., performing a numerical experiment. The resulting error estimate then becomes *a posteriori*, that is, after the experiment is performed.

Furthermore, we define an optimal mesh: A mesh is said to be optimal when the measure of error in the solution is equal for each element in the mesh. Those measures of error that are commonly used are the energy norm, relative percentage energy norm error, and local and global effectivity indices. The effectivity index, $\Theta = \|e\|/\|e\|_{ex}$, where $\|e\|$ and $\|e\|_{ex}$ are computed and exact errors, respectively, in the energy norm.

In Chapter 5, we point out that the results of this study suggest that the properties of a grid are mandated by the physical model, as well as the geometry. As Canonical Problems No. 2 and 3 of Chapter 3 showed, the gridding requirements for a TM problem can differ considerably from those for a CEM problem, even if the geometry of each problem is a simple rectangle. The distinction between the two problems and their meshes lies in the nature of the boundary conditions and the nonlinear behavior of the material parameters; i.e., it is the physical model that distinguishes the needs of each problem.

The major implementation problem in handling coupled analyses will be in supervising the transfer of data between the two meshes, and in adapting each mesh so that it becomes optimal for its respective problem domain. Little, apparently, has been done in the area of adaptive gridding for coupled problems, and we propose that Canonical Problem No. 3 could be the basis for a future research effort in this area. Even though the geometry for this problem is a simple rectangle, the physical models would present several areas where a grid would have to adapt in order to become optimal. The nonlinearity in the electrical conductivity might drive the grid adaptation, as well as the electrical singularity at the edge of the upper electrode. The thermal grid might have to adapt in response to the value of the convection coefficient. Canonical Problem No. 3 could also be the basis for further studies in *hp*-adaptive analysis of coupled problems.

2 CURRENT PRACTICES

This chapter raises the question of what features are necessary in a software package that efficiently solves coupled problems. The development of such software is an active field of study, and we undertook a survey of the literature in CEM to learn of this activity. The following is based on that survey.

In [1] a unified data-base (DB) for data exchange among field computation and CAE programs is presented. The DB consists of four units, by function:

1. The **pre-processing** unit contains:
   - a commercial solid modeling system called Geomod.
   - a commercial mesh generating system, Supertab.
   - an automatic mesh generator for electromagnetic field design.
   - an object manager which provides a set of operations for maniupulating data stored in the database. These operations enable information to be treated as *objects* instead of FORTRAN arrays. This frees users from the internal representation of objects, and enables designing at a high level of abstraction.
2. The **processing** unit contains three established field processors. They are:
   - Flux3d, which uses the finite element (FE) method.
   - Phi3d, which uses the boundary-integral equation (BIE) method.
   - Trifou, which uses a mixed FE-BIE method.
3. The **post-processing** unit is made up of two post-processors. Current efforts consist of developing a common post-processor for all three field codes.
4. The **data administration** unit is an interactive data-structuring module, through which the data administrator creates and maintains the basic TDS. (TDS is an abbreviation for a unified data structure called the TRIFLUX data structure, which is based on the characterization of operational data used in the three field codes. It contains geometrical data structure, finite-element data structure, inductors, and material properties.)

This type of system is useful in solving integrated problems, such as those involving electromagnetic, structural, and thermal variables. These are precisely the same problem one faces in designing microwave tubes. Clearly, the data administration unit, that was just described, is relevant to the questions that we are addressing here.

An alternative approach to handling the data base for computer modeling would be to store the geometry and material information in a "universal" data base to be accessed by all analysis programs. The model would be sliced by planes to create a series of cross sections. Each cross section would be stored as a bit map, with the "color" of each pixel indicating the material. Material properties, such as magnetic permeability, thermal conductivity, or modulus of elasticity would be stored in a table. Since meshing is dependent on the type of analysis performed, this information would not be included in the universal data base, but rather generated from it by one or more mesh preprocessors. Each analysis code would utilize a mesh preprocessor appropriate for the type of mesh it requires. While this approach may place a greater load on the mesh generator, it does provide a common model data base for all analysis programs.

The tool described in [2] utilizes the object oriented language, SMALLTALK, to allow descriptions of device classes, parameterizations and graphical-aided creations and modifications, use of part-whole hierarchy and multiple inheritance, in a highly uniform environment. It also serves as an interface to finite-element packages.

The work described in [1] has been extended recently to solving coupled problems that involve electromagnetic, mechanical and acoustic aspects [3]. The physical domains to be modeled span the range of electromagnetics to acoustics, and the solution algorithms include finite-elements (electromagnetics and mechanics), and boundary-integral equations (acoustics). A Supervisor FLYSTRYD is developed that links the three successive computations of electromagnetic, mechanical, and acoustic quantities. Electromagnetic problems are solved using FLUX3D, mechanical problems using SYSTUS, and acoustics using ASTRYD. See Figure 1. FLYSTRYD interpolates significant

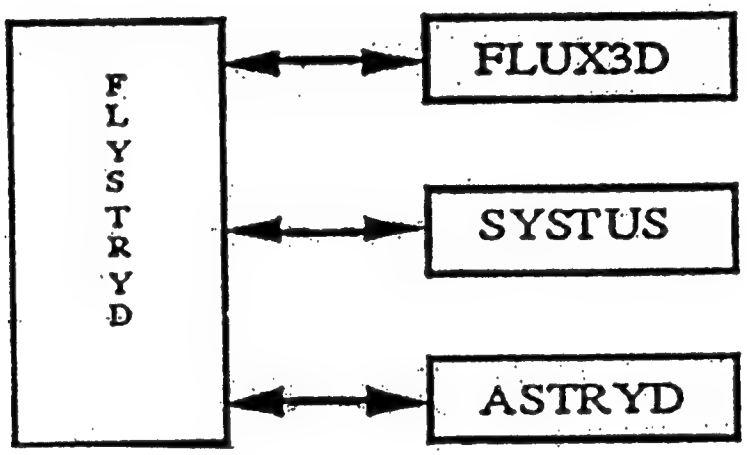


Figure 1: Relationship between the various programs.

physical quantities on different meshes by means of the shape functions that are defined on each mesh. The mesh generator of FLUX3D is provided to SYSTUS through FLYSTRYD. The reason for this is that electromagnetic mesh generators are generally more efficient for coupled problems, because they must be able to mesh the complex air regions surrounding the body of interest. A typical application for such mixed-modeling is the analysis of an oil-filled power inductor for the electric utility industry (Figure 2).

In Figures 3 and 4, we note that instead of just transferring the interesting data from the output files of one application software to the input files of another, the supervisor carries out that transfer between different meshes and geometries [3]. A consequence of this is that FLYSTRYD must handle two data-bases at the same time. Those data-bases contain two different models of the same physical problem—for instance, electromagnetic and mechanical models. This allows the supervisor to interpolate important physical quantities on different meshes. Interpolation is performed, as usual, by way of the shape functions of the finite elements. The difficulty arises because of the wide variety of types of elements of the different analysis codes. In this application, acoustic elements at the interface are mostly surface elements, while the electromagnetic mesh is composed of tetrahedrons. The electromagnetic mesh is usually composed of a great number of elements, so as to define correctly thin surfaces as, for instance, air gaps. Because the mechanical problem uses more nodal unknowns than does the electrical problem, the number of elements of

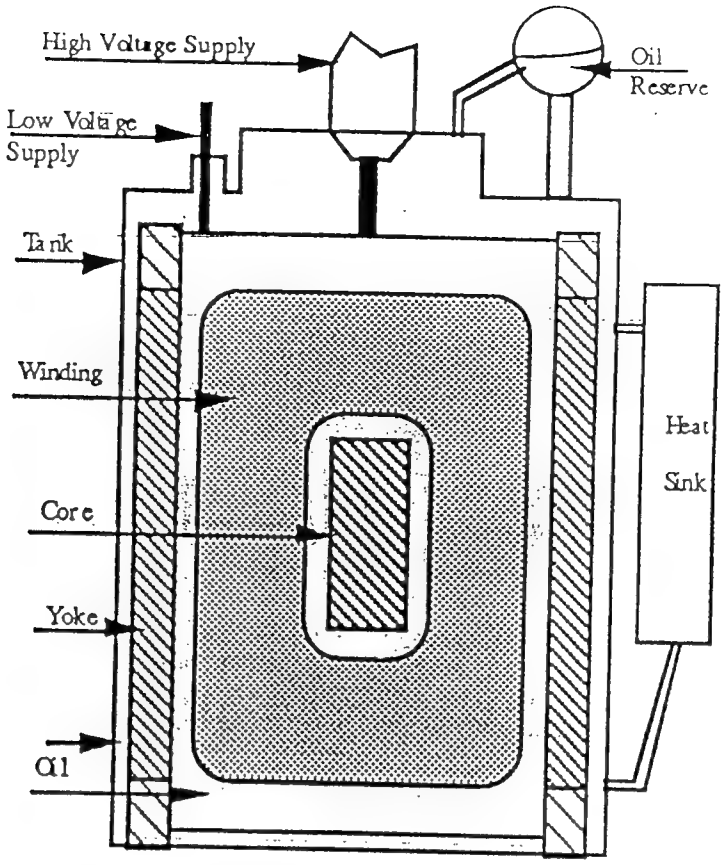


Figure 2: An oil-filled power inductor for the electric utility industry.

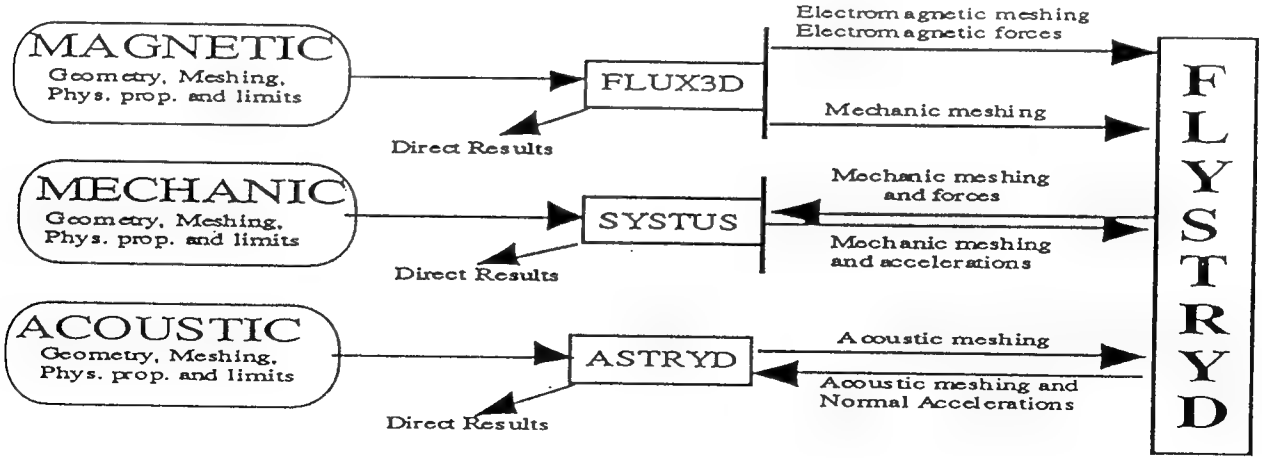


Figure 3: A view of FLYSTRYD.

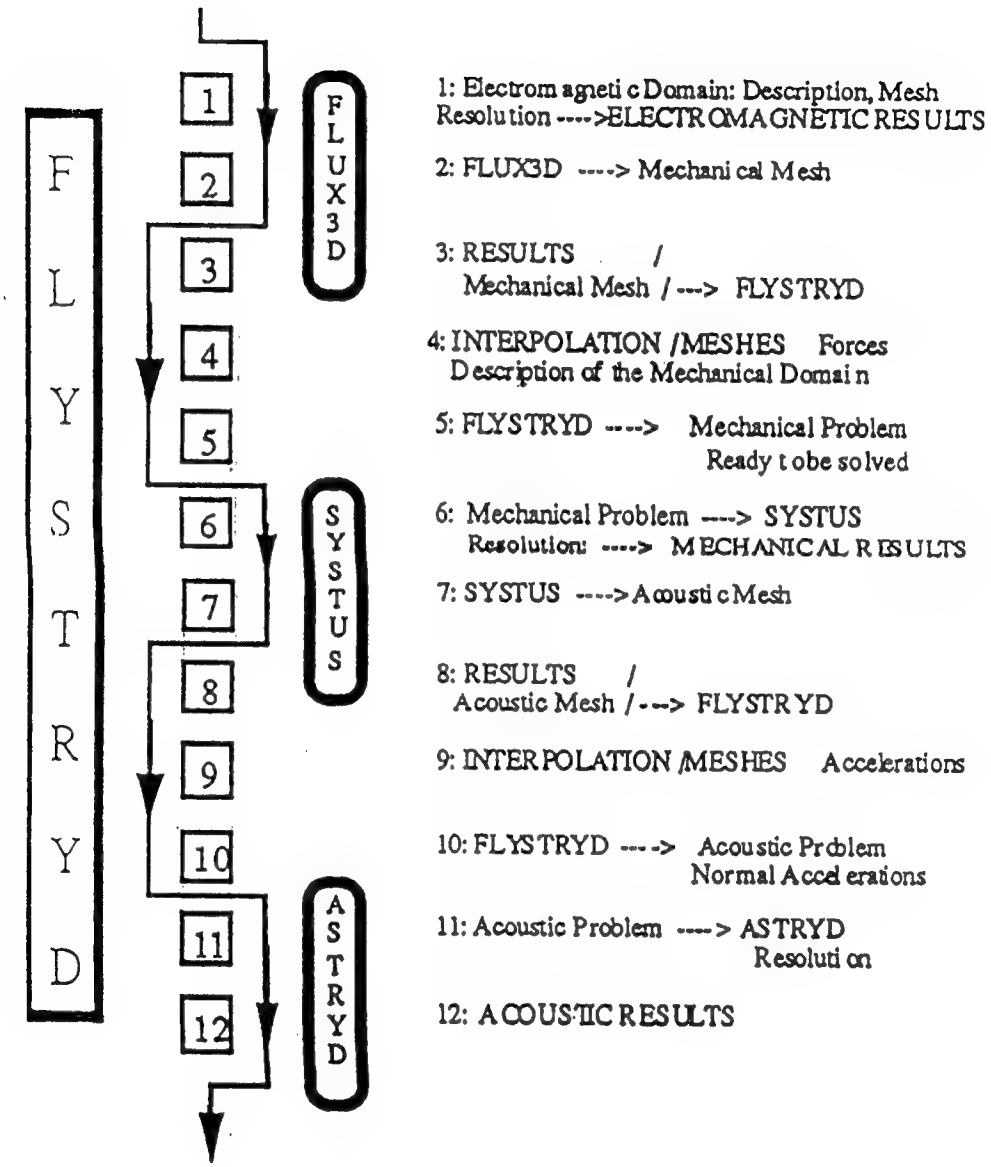


Figure 4: A closer look at FLYSTRYD.

the mechanical mesh is smaller [3].

The authors of [3] stress that the mechanical mesh is provided to SYSTUS by FLUX3D through FLYSTRYD. Though the mechanical and electrical meshes are different, the mesh generator of FLUX3D appeared to be more efficient when confronted with a complex electromagnetic structure as occurs in the power inductor problem (recall Figure 2). This is generally true, because electromagnetic finite-element meshes must be able to mesh the air domain, which may have a very complex shape.

When the geometries of the various physical models differ, one has to use projection or extrapolation techniques to provide the correct data. Projection is used when a physical quantity is defined on a source region which does not exist in the target model. On the other hand, extrapolation is necessary to define a quantity in a region that did not exist in the source domain. The inductor provides a good example; in the mechanical analysis of [3], the cooling and rigidification armature of the tank was modeled with linear elements. In the acoustic model, however, the same physical structure had to be modeled by a surface of complex shape, because those cooling elemetns proved to be important for the acoustic elements. Transmission of normal acceleration from the simplified flat tank surface of the mechanical domain to the actual shape of the tank is a good example of the use of extrapolation techniques [3].

A similar problem is faced in other applications of coupled electromagnetic and mechanical problems. In [4], in which one computes mechanical forces of electromagnetic origin, the load on the vacuum vessel was obtained using an electromagnetic finite-element code that utilized 763 triangular elements, with 442 nodes. The finite-element model for the mechanical stress analysis, however, used a finer mesh, with different elements. This required that an interface code be developed that would allow the calculated electromagnetic loads to be applied to the stress analysis model.

The main feature of this code is a load transfer from a point located on one model to the geometrically closest point on another model. This approach is enhanced in order to control some integral characteristics of a load field, such as average forces and moments. The mapped load field is adjusted in such a way that average forces and moments are coincident, both for the entire models and for the same characteristic part of the models. The difference in the force component before and after interpolation does not exceed 3.4%, and that for the moments is about 2.7% [4].

3 ANALYSIS OF CANONICAL PROBLEMS

3.1 Introduction

This chapter deals with the central theme of this research, which is to perform numerical experiments on coupled thermal-electromagnetic problems, and draw conclusions from those experiments. Thus, the first question to be raised was to determine suitable canonical problems that would provide those answers that we sought. We chose to emphasize thermal-EM problems, rather than other coupled problems, because these problems are typical of those that are met in designing high-power analog circuit modules, such as T/R modules. Furthermore, the problems chosen could be solved entirely through the use of one finite-element code, namely TOPAZ, a three-dimensional LaPlace-Poisson solver that was developed at the Lawrence Livermore National Laboratory.

The problems had to be simple enough that important features would not be obscured by unnecessary complexities, yet rich enough to display those features that we decided were important to the research.

3.2 Definition of the Problems

3.2.1 Problem 1: Three-dimensional heat-flow with a temperature-dependent thermal conductivity

Problem 1, which is illustrated in Figure 5, is based on a model of heat-flow in T/R modules and

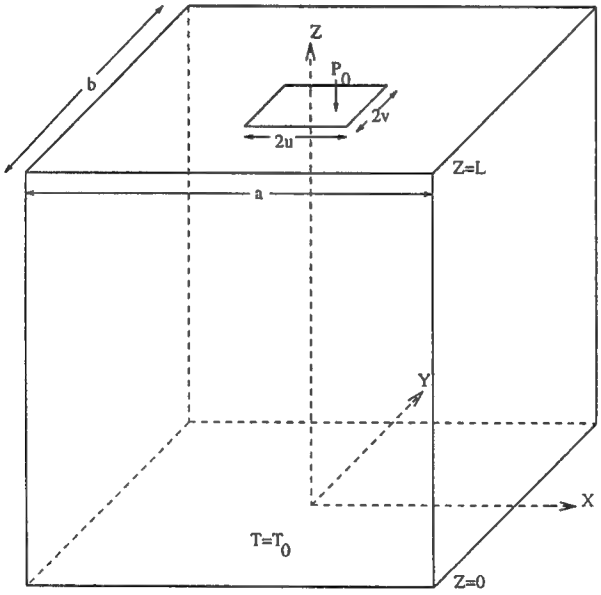


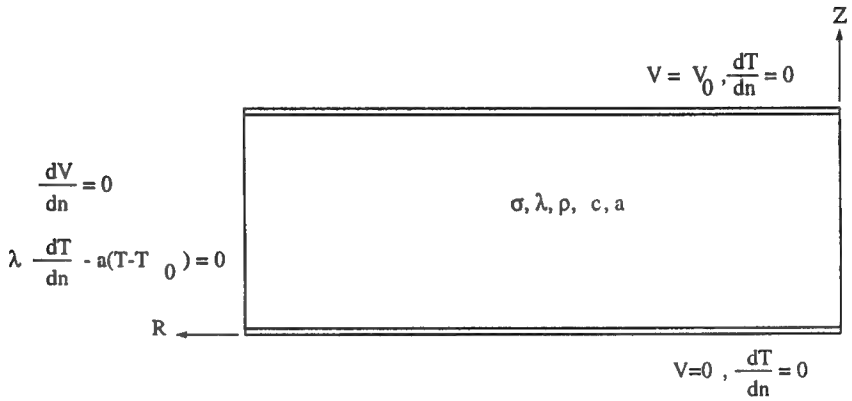
Figure 5: Three-dimensional heat-flow in a body with a temperature-dependent thermal conductivity. The lateral surfaces are insulated, while the bottom is maintained at temperature T_0, and the top receives power, P_0, through the power dissipation surface, whose area is $4uv$.

multichip modules. The four lateral surfaces of a block of material are maintained under adiabatic conditions, while the distribution of heat-flux is defined over the top surface (a 'generalized adiabatic condition'), and the bottom surface is maintained at temperature T_0. The Z-dimension is greatly exaggerated in the figure. Though it is not a coupled problem, it was chosen because we were able to compute an analytic solution to it, even in the case in which the thermal conductivity depends upon

the unknown temperature which makes it nonlinear. The thermal conductivity versus temperature profile is chosen to be representative of a number of materials that are of interest to T/R modules. This problem also allows us to validate TOPAZ's ability to solve nonlinear problems in three-dimensions, and allowed us to become familiar with its use.

3.2.2 Problem 2: A coupled CEM and thermal problem in free-space

Problem 2, which is shown in Figure 6, is a simplification of the problem studied in [8]. It is basically an 'infinite', parallel-plate, axisymmetric capacitor, whose dielectric is barium titanate $BaTiO_3$. The conductivity of $BaTiO_3$ is a function of temperature and electric field, as illustrated in Figure 7 [8]. Hence, the problem is nonlinear. This, together with the fact that the geometry is quite simple, and that both the thermal and electrical parts of the problem can be solved using TOPAZ are the reasons for choosing this problem and the next one.



$$\nabla \cdot (\sigma \nabla V) = 0$$

$$\nabla \cdot (\lambda \nabla T) + \rho c \frac{dT}{dt} = \sigma (\nabla V)^2$$

$$\sigma = \sigma(T,E)$$

Figure 6: An axisymmetric coupled CEM and thermal problem. The electrical conductivity is a function of temperature and electric field, which makes the coupled problem nonlinear.

The material constants are as follows:

Material Constants for Barium Titanate [8]

σ (S/m)	λ (W/mK)	$c\rho$ (J/m^3K)	a (W/m^2K)
—	4.5	3.00×10^6	20

and the electrical conductivity as a function of temperature and electric field is shown in Figure 7.

The meshes for the EM and TM parts will probably be distinct; this is especially so if the thermal problem is modeled as having a conducting lateral surface, instead of the convective boundary condition shown in Figure 6, or if the convection coefficient, a, is large. The mesh for the thermal

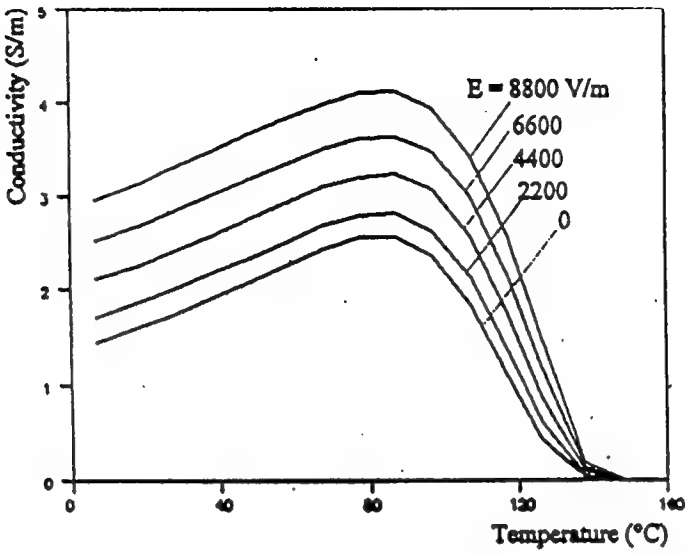


Figure 7: Electrical conductivity of Barium Titanate as a function of electric field and temperature [8].

problem in the former case must extend to infinity, with Dirichlet conditions on the temperature at infinity. That is, the temperature is specified far from the lateral surfaces.

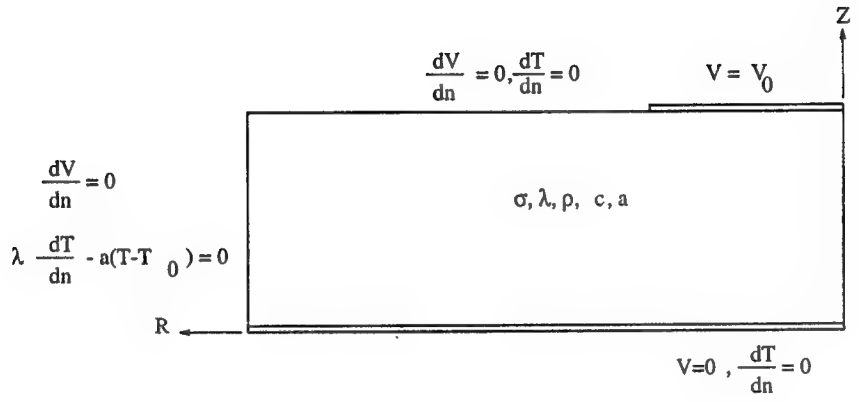
The thermal boundary conditions shown in Figure 6 correspond to having having the top and bottom surfaces insulated, and the lateral surface cooled by convection, with the convective coefficient equal to a. The convection boundary condition is often referred to as a mixed boundary condition of the third kind. The condition at the radius, $r = 0$, is that there is no heat flux in the radial direction. The larger a is, the steeper will be the thermal gradient in the radial direction, and this will have an effect on the meshing requirements for the thermal problem, even though the electrical problem will have a rather uniform field, if the capacitor's height is small compared to its radius.

Homogeneous (Neumann) boundary conditions, in which the normal derivative is specified, indicate that the surface is insulated, in the case of a thermal problem, and that there is no electric current flowing through that surface, in the case of an electric problem. Hence, these distinct boundary conditions for the electric and thermal aspects of the problem, mean that the mesh generator-preprocessor must keep track of those parts of the mesh that are used to solve the thermal problem and those used to solve the electromagnetic problem.

This problem requires only a Poisson (or Laplace) solver to compute the electric and thermal fields; hence, we can use TOPAZ3D to do the entire coupled problem.

3.2.3 Problem 3: Another coupled CEM and thermal problem in free-space

Problem 3 differs from Problem 2 in that the electrical boundary conditions on the top surface are more complicated. See Figure 8. All other conditions of the problem are identical to Problem No. 2. If we ignore any nonlinearities in Canonical Problem No. 2, then the electric field throughout the body is virtually uniform, if the radius is much larger than the height; i.e., there is little fringing. This means that the thermal loading is also virtually uniform. In Canonical Problem No. 3, however, we expect that the thermal and electrical meshes will both be more complicated than the meshes in Problem No. 2; the grading of the meshes will be more severe near the top electrode.



$$\nabla \cdot (\sigma \nabla V) = 0$$

$$\nabla \cdot (\lambda \nabla T) + \rho c \frac{dT}{dt} = \sigma (\nabla V)^2$$

$$\sigma = \sigma(T,E)$$

Figure 8: An axisymmetric coupled CEM and thermal problem. This problem differs from Canonical Problem No. 2, in that the electrical boundary condition on the top surface is more complicated, due to the fact that the electrode does not cover the entire surface.

3.3 TOPAZ

TOPAZ3D [7], a three-dimensional, implicit, finite-element code for solving thermal problems (or general Laplace-Poisson equations), was applied to all three problems. There are a number of features of TOPAZ3D that make it useful for our problems:

- it solves for steady state or transient temperature fields on three dimensional geometries
- material properties may be temperature dependent and isotropic or orthotropic (three mutually perpendicular planes of symmetry)
- time and temperature dependent boundary conditions can be specified: temperature, flux, convection, and radiation

TOPAZ3D solves the differential equation of heat-flow (generalized Poisson equation):

$$\rho c \frac{\partial \theta}{\partial t} = \frac{\partial}{\partial x}\left[k_x \frac{\partial \theta}{\partial x}\right] + \frac{\partial}{\partial y}\left[k_y \frac{\partial \theta}{\partial y}\right] + \frac{\partial}{\partial z}\left[k_z \frac{\partial \theta}{\partial z}\right] + q_g \quad \text{in } \Omega \,, \tag{1}$$

subject to the boundary condition:

$$k_x \frac{\partial \theta}{\partial x} n_x + k_x \frac{\partial \theta}{\partial x} n_x + k_x \frac{\partial \theta}{\partial x} n_x + \beta\theta = \gamma \quad \text{on } \Gamma \,, \tag{2}$$

and initial condition (transient problems):

$$\theta = \theta(x, y, z) \quad \text{at } t = t_0 \tag{3}$$

The TOPAZ3D finite element discretization uses an 8-node trilinear hexahedral element, that can degenerate into a 6-node triangular prism, or a 4-node tetrahedron, as shown in Figure 9

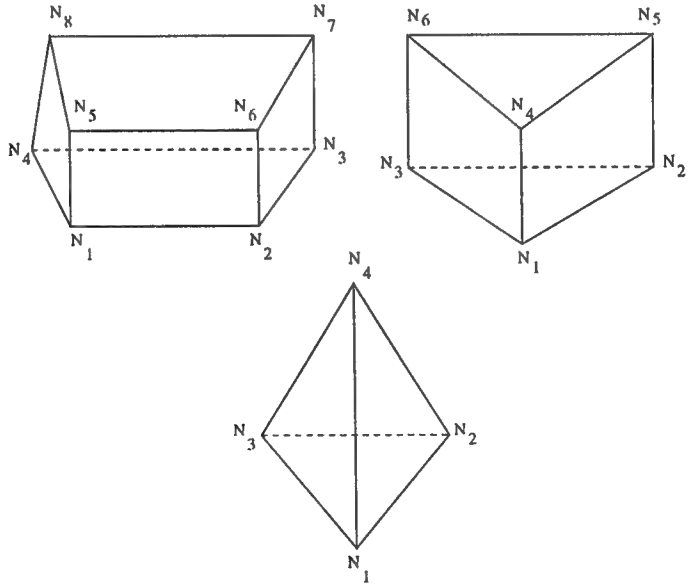


Figure 9: Elements used in the TOPAZ3D finite-element discretization.

3.4 An Algorithm for Solving Coupled Nonlinear Problems

An algorithm for solving the coupled nonlinear problems 2 and 3 is presented in Figure 10, and is explained by the following steps:

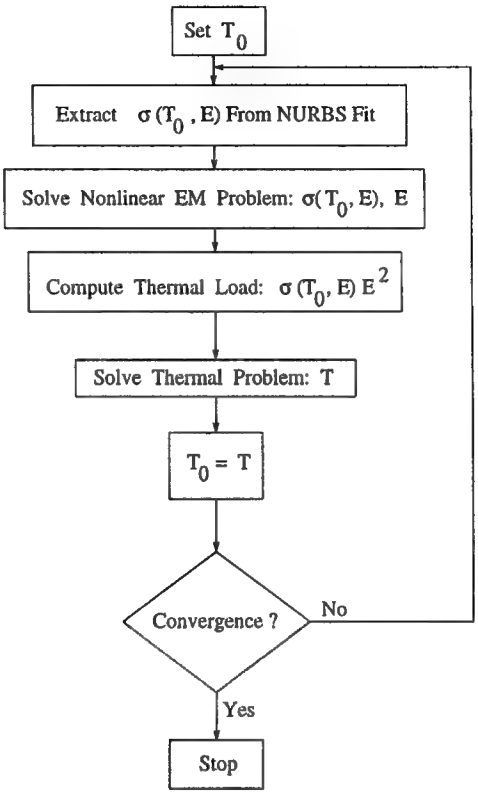


Figure 10: An algorithm for solving coupled nonlinear CEM/TM problems.

1. **Data Fitting:** Using a constant value of temperature, perform a NURBS (Non-uniform Rational B-Spline) fit to the conductivity data as a function of electric field.

2. **Electrical Problem:** Solve, $\nabla \cdot (\sigma \nabla V) = 0$, with a nonlinear conductivity, that depends only upon the electric field: $\sigma = \sigma(\mathbf{E})$.

3. **Thermal Problem:** Using $\sigma(\nabla V)^2$ as the distributed thermal load, compute the temperature, T. Use constant values for λ, ρc, and a, as given in the table of material constants.

Canonical Problem No. 1

Solution of the problem. We use the transformation

$$\theta(x,y,z) = \int_{T_0}^{T(x,y,z)} k(T)dT \ , \tag{4}$$

which yields Laplace's equation

$$\nabla^2\theta = 0 \ . \tag{5}$$

Note that the equation for θ is valid regardless of the nature of the thermal coefficient, $k(T)$. LaPlace's equation for temperture is valid only in a region of uniform, linear, and isotropic thermal coefficient.

We will apply (5) to an infinite slab, rather than to the canonical problem shown in Figure 5. In this problem we require that $\theta(x,y,0) = 0$, and $\partial\theta(x,y,z)/\partial z|_{z=L}$ = heat flux at $z = L = j_0 4uv[\sin(k_x u)/k_x u][\sin(k_y v)/k_y v]$.

The two-dimensional spatial transform of θ satisfies $d^2\tilde{\theta}/dz^2 - l^2\tilde{\theta} = 0$. The solution of this equation is $\tilde{\theta}(k_x,k_y,z) = \tilde{A}\exp(lz) + \tilde{B}\exp(-lz)$. The boundary conditions are expressed in terms of the transform coefficients as

$$\begin{aligned} \tilde{A} + \tilde{B} &= 0 \\ \tilde{A}e^{lL} - \tilde{B}e^{-lL} &= -j_0\frac{4uv}{l}\frac{\sin(k_x u)}{k_x u}\frac{\sin(k_y v)}{k_y v} \ . \end{aligned} \tag{6}$$

The solution of (6) is

$$\begin{aligned} \tilde{A} &= -\frac{j_0\dfrac{4uv}{l}\dfrac{\sin(k_x u)}{k_x u}\dfrac{\sin(k_y v)}{k_y v}e^{-lL}}{1+e^{-2lL}} \\ \tilde{B} &= \frac{j_0\dfrac{4uv}{l}\dfrac{\sin(k_x u)}{k_x u}\dfrac{\sin(k_y v)}{k_y v}e^{-lL}}{1+e^{-2lL}} \ , \end{aligned} \tag{7}$$

which means that

$$\tilde{\theta}(k_x,k_y,z) = \frac{j_0 4uv\dfrac{\sin(k_x u)}{k_x u}\dfrac{\sin(k_y v)}{k_y v}}{1+e^{-2lL}}\left[\frac{e^{-l(z+L)} - e^{-l(L-z)}}{l}\right] \ . \tag{8}$$

Note that $\tilde{\theta}(l,z) \to -j_0 4uvz$ as $l \to 0$. The inverse transform produces the final result

$$\theta(x,y,z) = \frac{j_0 4uv}{4\pi^2}\int\int_{-\infty}^{\infty}\frac{\dfrac{\sin(k_x u)}{k_x u}\dfrac{\sin(k_y v)}{k_y v}}{1+e^{-2lL}}\left[\frac{e^{-l(z+L)} - e^{-l(L-z)}}{l}\right]e^{-j(k_x x + k_y y)}dk_x dk_y \ . \tag{9}$$

A **one-dimensional model** results if the heat-flux source is applied uniformly over the entire surface, $z = L$. In that case, the transform of the heat-flux approaches $j_0 4\pi^2\delta(k_x)\delta(k_y)$ as $(u,v) \to (\infty,\infty)$. Then the solution, (9), becomes

$$\theta(z) = -j_0 z \ . \tag{10}$$

A **finite-dimensional model** is obtained when the lateral dimensions of the chip are bounded in the (x, y)-plane. If the length in the x-direction is a, and that in the y-direction is b, then the solution that corresponds to (9) is the Fourier series given by

$$\theta(x,y,z) = \frac{j_0 4uv}{ab} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \frac{\frac{\sin(\frac{2m\pi}{a}u)}{\frac{2m\pi}{a}u} \frac{\sin(\frac{2n\pi}{b}v)}{\frac{2n\pi}{b}v}}{1+e^{-2lL}} \left[\frac{e^{-l(z+L)} - e^{-l(L-z)}}{l}\right] e^{-j2\pi(\frac{m}{a}x + \frac{n}{b}y)} , \tag{11}$$

where $l^2 = 2\pi^2 \left[(m/a)^2 + (n/b)^2\right]$. Note the similarity between (9) and (11) when $dk_x = 2\pi/a$ and $dk_y = 2\pi/b$. In fact, the use of a Fourier series representation over a bounded domain of the Fourier integral, (9), is precisely the idea behind the FFT. The total heat input is given by $P_0 = j_0 4uv$.

a. Application to a Linear Problem. After evaluating $\theta(x, y, z)$ from (9), we can solve linear and nonlinear heat-flow problems by using (4). First, we demonstrate the procedure on a linear problem, in which k is independent of temperature.

The integral in (4) yields

$$\theta(x,y,z) = \int_{T_0}^{T(x,y,z)} k(T)dT = k\left[T(x,y,z) - T_0\right] , \tag{12}$$

from which we get

$$T(x,y,z) = T_0 + \frac{\theta(x,y,z)}{k} . \tag{13}$$

b. Application to a Nonlinear Problem. A number of materials of interest to T/R modules have a nonlinear thermal coefficent, which can generally be modeled over a limited temperature range as

$$k(T) = \frac{K_0}{T - T_0} , \tag{14}$$

where K_0 and T_0 are characteristic for each material [5] [6]. (Do not confuse T_0 here with the temperature of the thermal bath in Figure 5.) For example, $K_0 = 32$ W/mm, $T_0 = 80$K for silicon.

Hence,

$$\begin{aligned} \theta(x,y,z) &= \int_{T_a}^{T_b} k(T)dT \\ &= K_0 \ln\left|\frac{T_b - T_0}{T_a - T_0}\right| , \end{aligned} \tag{15}$$

from which we get

$$T_b = T_0 + (T_a - T_0)e^{\theta(x,y,z)/K_0} . \tag{16}$$

The solution for the temperature in Figure 5 is, therefore,

$$T(x,y,z) = T_{01} + (T_0 - T_{01}) \exp(\theta(x,y,z)/K_{01}) , \tag{17}$$

where T_{01} and K_{01} are the two parameters that appear in (14).

We have computed the temperature distribution for Figure 5, using the following parameter values (lengths in meters, temperatures in degrees K):

$$\begin{aligned} a = b &= 6.35 \times 10^{-3} \\ L &= 2.24 \times 10^{-3} \\ z &= 2.1 \times 10^{-3} \\ P_0 &= -6.05 \text{ Watts} \\ T_0 &= 300°\text{K} \\ T_{01} &= 80°\text{K} \\ K_{01} &= 3.2 \times 10^4 \text{ Watts/meter} \end{aligned} \tag{18}$$

Results for values of x along the line $(y = 0, z = 2.1 \times 10^{-3})$ and for various values of the power-dissipation surface parameters, (u, v), are given in Table 1.

	$T(x)$		
x	$u = v = 2.112 \times 10^{-3}$	$u = v = 0.5 \times 10^{-3}$	$u = v = 0.0$
0	0.30361E+03	0.31701E+03	0.35044E+03
2.48×10^{-4}	0.30360E+03	0.31583E+03	0.32245E+03
4.96×10^{-4}	0.30357E+03	0.31186E+03	0.31127E+03
7.44×10^{-4}	0.30351E+03	0.30751E+03	0.30703E+03
9.92×10^{-4}	0.30342E+03	0.30513E+03	0.30488E+03
12.4×10^{-4}	0.30330E+03	0.30374E+03	0.30360E+03
14.9×10^{-4}	0.30314E+03	0.30268E+03	0.30277E+03
17.4×10^{-4}	0.30291E+03	0.30227E+03	0.30220E+03
19.8×10^{-4}	0.30258E+03	0.30185E+03	0.30181E+03
22.3×10^{-4}	0.30215E+03	0.30156E+03	0.30153E+03
24.8×10^{-4}	0.30184E+03	0.30137E+03	0.30134E+03
27.3×10^{-4}	0.30165E+03	0.30124E+03	0.30121E+03
29.8×10^{-4}	0.30155E+03	0.30117E+03	0.30114E+03

Table 1: Solution of Problem No. 1 for various values of the power-dissipation surface parameters, (u, v).

c. Solution via TOPAZ We used symmetry to reduce the problem to one-fourth of its original size, and introduced a grid consisting of $13 \times 13 \times 5$ $(X \times Y \times Z)$ nodes. This served as a test of the ability of TOPAZ3D to solve nonlinear problems, as will be required in the next problem.

TOPAZ3D allows a database consisting of eight pairs of temperature versus thermal conductivity entries to be used for the interpolation that is required in solving nonlinear problems. Furthermore, it requires an initial temperature to start the iteration process. We found that the solution of this problem was rather insensitive to the starting temperature, but was much more sensitive to the distribution of the database entries, as we would suspect, since the quality of the interpolation of k versus T depends on that distribution.

In our first exercise, we used the following data for the database, which was derived from (14):

T	$k(T)$
300° K	145.45
305	142.22
310	139.13
315	136.17
320	133.33
325	130.61
330	128.00
335	125.49

In Table 2, we compare the TOPAZ results with the analytic results for $u = v = 0.5 \times 10^{-3}$. The initial value for the temperature iterations is 317°K.

Table 2: Comparison of TOPAZ and Analytic Results.

X	Y	Z	Node#	Topaz (T)	Analytical Results (T)
0	0	2.1E-03	424	316.61	317.01
0.5E-3	0	2.1E-03	489	311.61	Don't have this point (311.79 via interpolation)
0.992E-3	0	2.1E-03	554	305.08	305.13
1.49E-3	0	2.1E-03	619	302.78	302.68
1.98E-3	0	2.1E-03	684	301.84	301.85
2.48E-3	0	2.1E-03	749	301.34	301.37
3.175E-3	0	2.1E-03	814	301.13	Don't have this point

In a second exercise, we extended the $T_1 \rightarrow T_8$ temperature range, and derived the following values for the database:

T	$k(T)$
90° K	3200.
180	320.
280	160.
380	106.67
480	80.
580	64.
680	53.33
780	45.71

Clearly, the spread in the values of $k(T)$ is much greater than in the first exercise, and the computed results (with a starting temperature of 325°K) are shown in Table 3.

The greatest discrepancy between the analytical and TOPAZ results occurs near the heat source, as we might suspect. The difference in the results of these two exercises is due to the fact that in the second case we are interpolating values of the thermal conductivity over a much larger range than in the first case, and this leads to slightly less accurate values for the conductivity.

The three-dimensional heat flow problem that was just solved analytically and by means of TOPAZ3D, while interesting in its own right, merely served to establish the ability of TOPAZ3D to solve nonlinear LaPlace or Poisson problems. The fact that we had an analytical benchmark to compare the TOPAZ solution was of extreme importance for the validation effort. The next two

Table 3: Comparison of TOPAZ and Analytic Results.

X	Y	Z	Node#	Topaz (T)	Analytical Results (T)
0	0	2.1E-03	424	316.076	317.01
0.5E-3	0	2.1E-03	489	311.254	Don't have this point (311.79 via interpolation)
0.992E-3	0	2.1E-03	554	304.937	305.13
1.49E-3	0	2.1E-03	619	302.700	302.68
1.98E-3	0	2.1E-03	684	301.789	301.85
2.48E-3	0	2.1E-03	749	301.306	301.37
3.175E-3	0	2.1E-03	814	301.099	Don't have this point

problems, however, are the crux of our numerical experiments during this project, and neither of them has an analytic solution. Nevertheless, we can apply TOPAZ3D to them with confidence, since we have already validated TOPAZ3D's ability to solve nonlinear problems.

Canonical Problem No. 2

a. Analysis of Uncoupled CEM and TM Problems. The solution of the uncoupled thermal and electromagnetic problems, even in the absence of any nonlinearities, will give us considerable insight into the gridding requirements. This is due to the fact that the convection coefficient a will play a significant role in the thermal problem, but not the electrical one, and as a result the grid for the thermal problem may differ considerably from the grid for the electrical problem, even though the geometry, which is quite simple, is identical for both problems. We will take the radius of the disk to be 10mm, and the height to be 2.5mm in the numerical experiments.

Electrical problem. Let $V_0 = 10$ V in Figure 6; then the potential distribution and electric field are shown in Figures 11 and 12. These are the solutions that one would expect for an 'infinite,' plane-parallel capacitor.

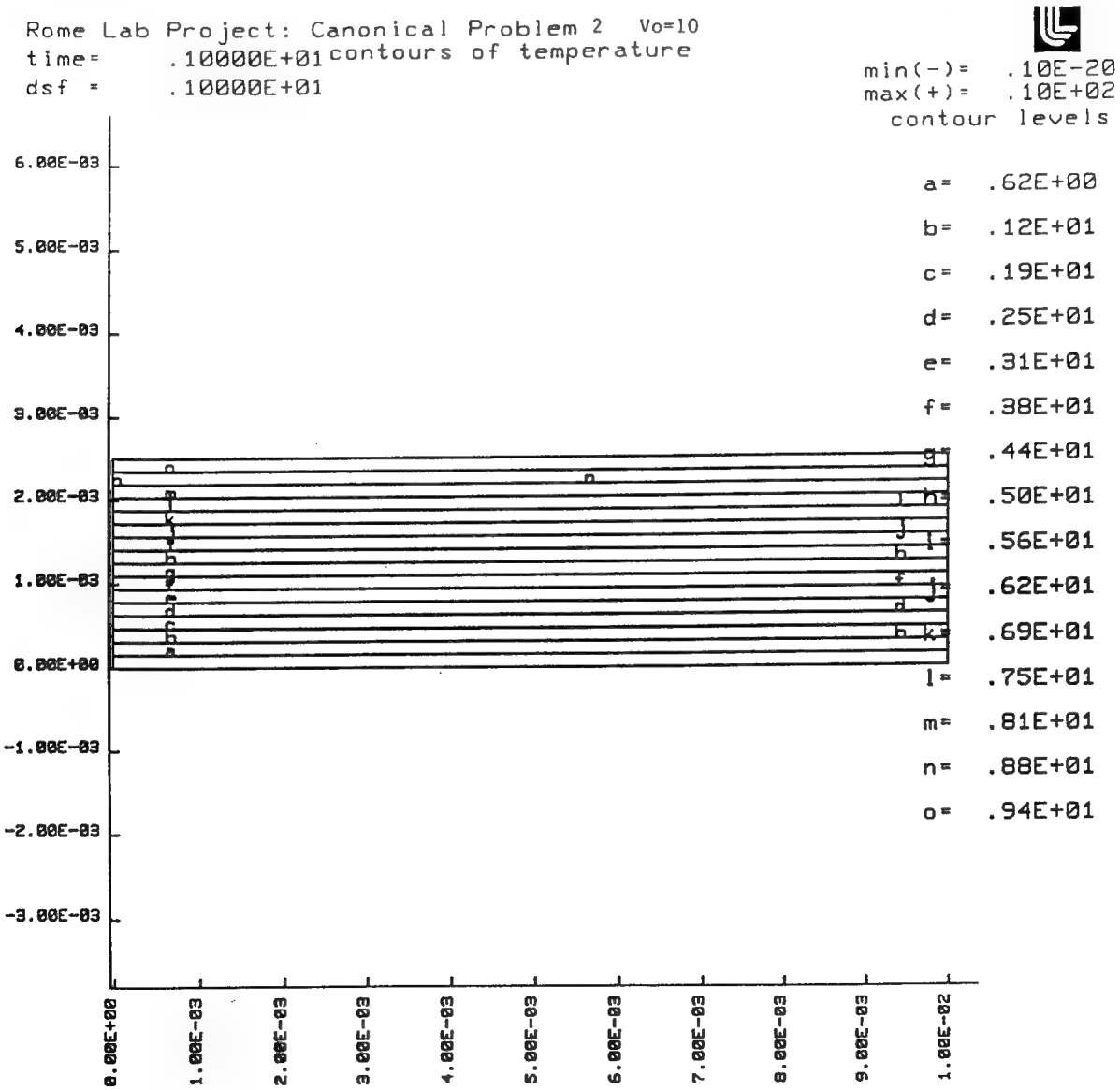


Figure 11: Potential distribution for Canonical Problem No. 2, when the electric and thermal problems are uncoupled.

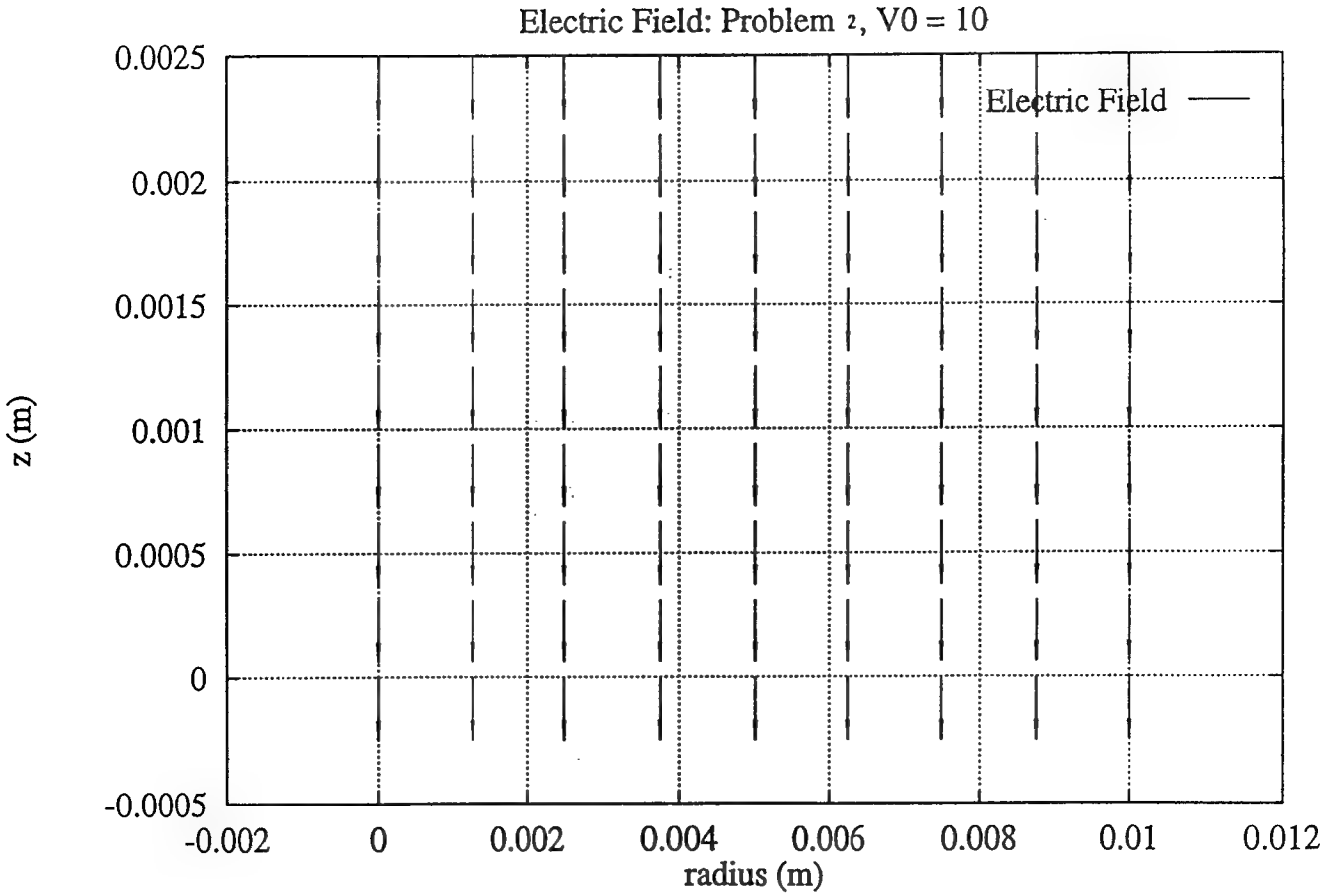


Figure 12: Electric field distribution for Canonical Problem No. 2, when the electric and thermal problems are uncoupled.

Thermal problem. There are a number of interesting cases to consider for the thermal problem. Unlike the electromagnetic problem, the gradients for the thermal problem are predominately in the radial direction, and, as will be seen, are strongly dependent upon the convective heat-transfer coefficent, a. The ambient temperature $T_0 = 20°$ in all of the thermal problems, and we use $\sigma(\nabla V)^2$ as the thermal load, where V is obtained from the electric calculation (this is the only degree of coupling in these otherwise uncoupled problems).

In Figure 13, we see the isothermal contours under the condition that $V_0 = 1V$, $a = 20$. Note that the relative thermal gradient in the radial direction is rather modest.

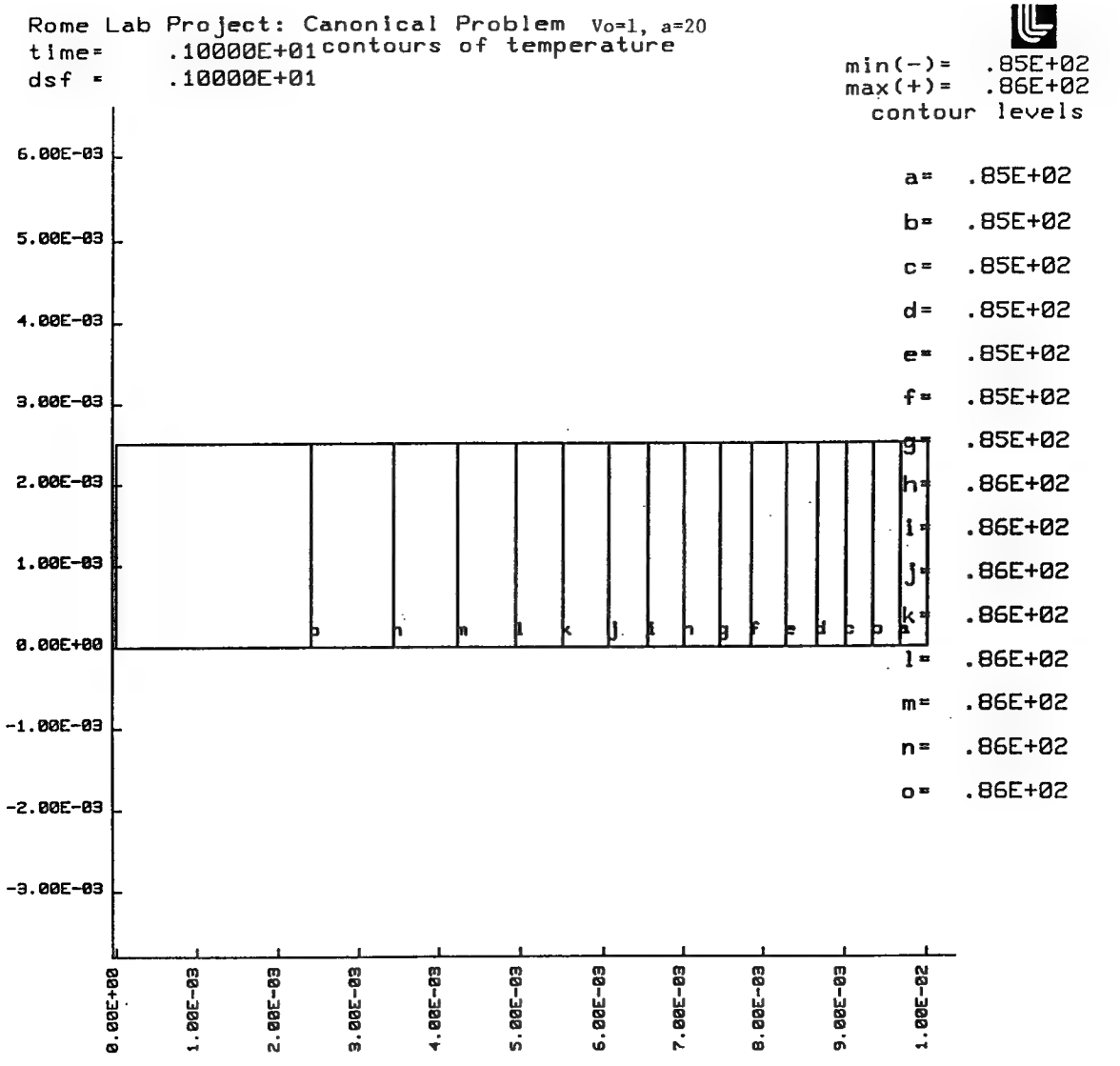


Figure 13: Isothermal contours when $V_0 = 1V$, $a = 20$. The right-hand edge is the exposed surface at which convection takes place.

When the convection coefficient is increased to $a = 200$, then the isothermal contours become those shown in Figure 14. Clearly, convection is playing a more significant role in cooling the slab. The relative temperature gradient remains modest, but is three-times as large as in Figure 13.

The more interesting results seem to occur with a larger electrical excitation, which results in a

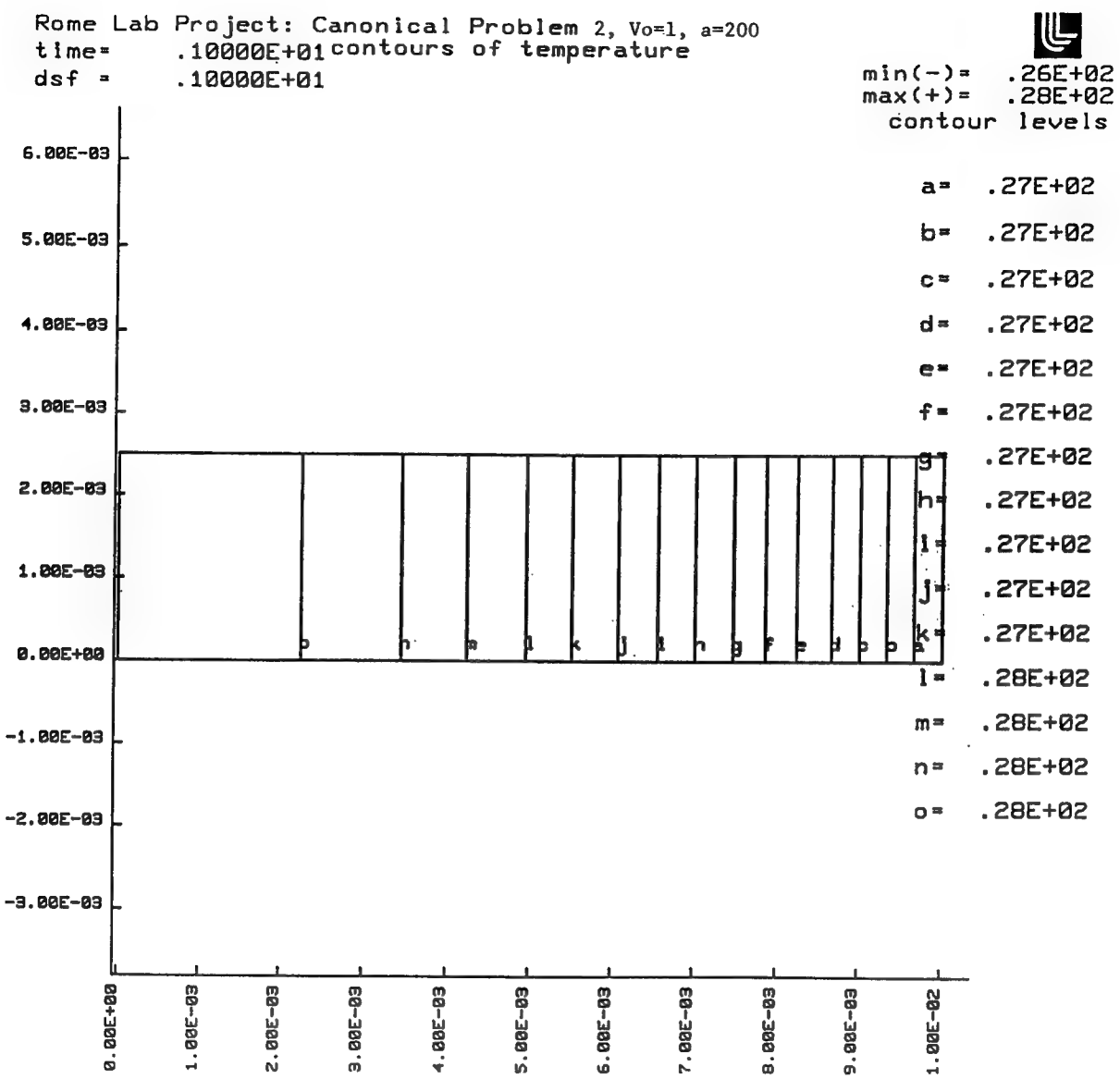


Figure 14: Isothermal contours when $V_0 = 1V$, $a = 200$. The right-hand edge is the exposed surface at which convection takes place.

larger thermal loading. For example, consider the results of Figure 15, for which $V_0 = 10V$, $a = 20$. The temperatures are about 100 times larger than those of Figure 13, as we would expect for a

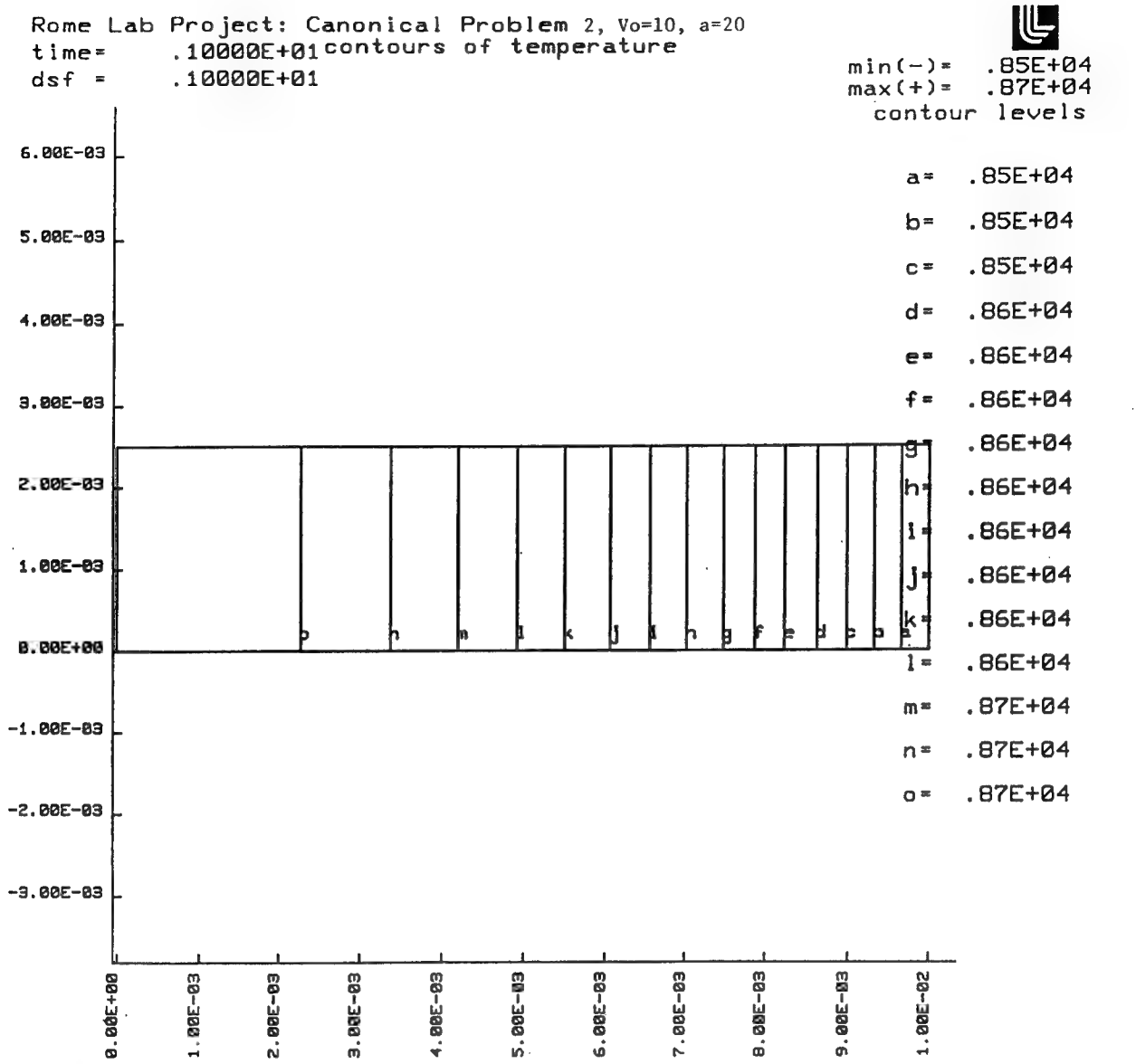


Figure 15: Isothermal contours when $V_0 = 10V$, $a = 20$. The right-hand edge is the exposed surface at which convection takes place. The temperatures are about 100 times larger than those of Figure 13.

linear problem in which the excitation is 100 times larger than its original value. The gradient, however, is still rather modest.

The final result for this series of tests is shown in Figure 16. The parameters for this problem are $V_0 = 10V$, $a = 200$. This combination of parameters produces a significant radial thermal gradient, and could require a finer mesh in the radial direction.

These tests indicate that, even though the geometry is quite simple, there is a significant difference in the grid requirements between the electrical and thermal problems. Indeed, the electrical gradients are in the z-direction, whereas the thermal gradients are in the radial direction. Furthermore, the thermal gradients are determined by the convection coefficient, a, which controls the

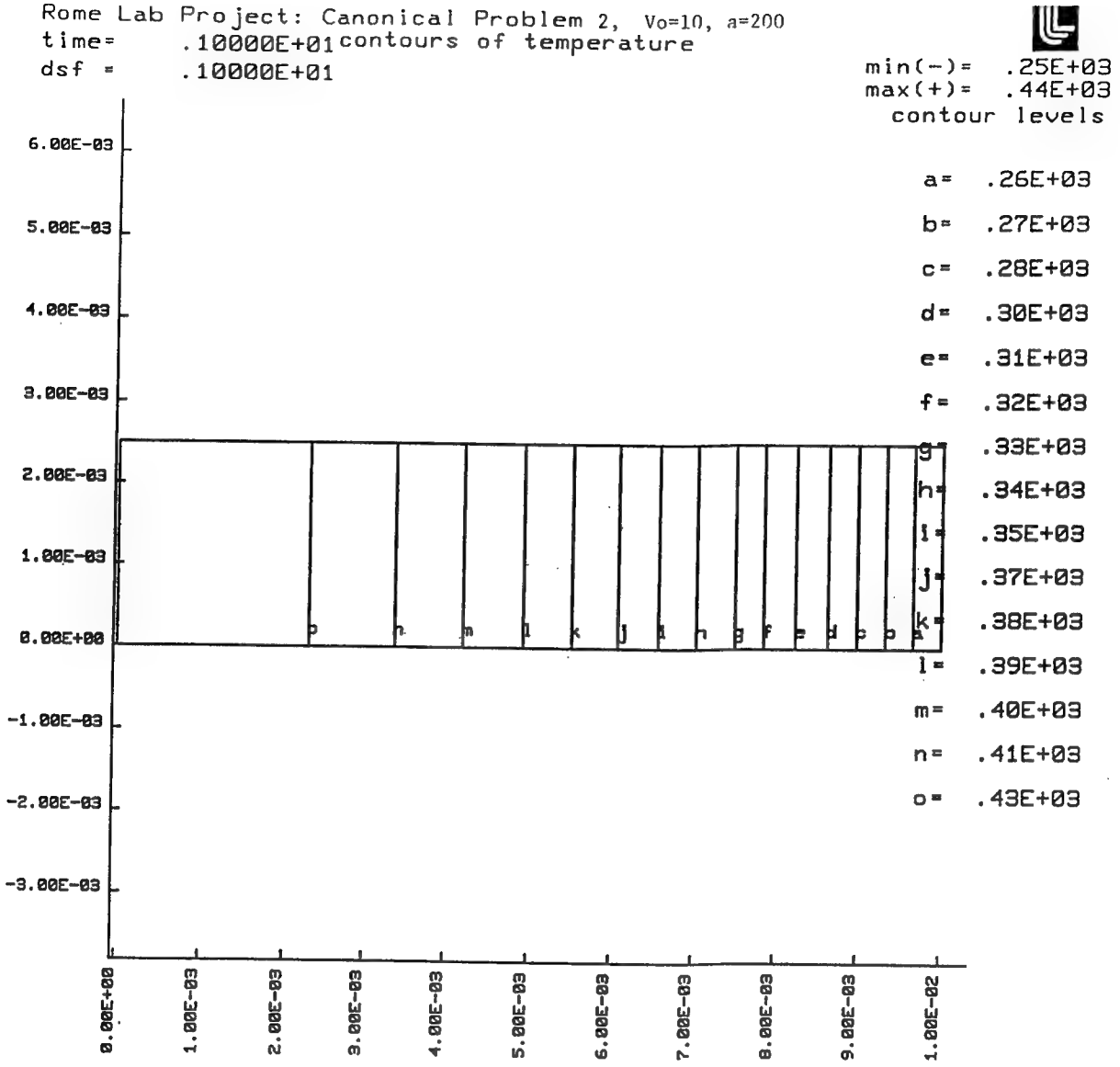


Figure 16: Isothermal contours when $V_0 = 10V$, $a = 200$. The right-hand edge is the exposed surface at which convection takes place. These parameters produce a significant thermal gradient.

thermore, the thermal gradients are determined by the convection coefficient, a, which controls the radial boundary condition. To put it another way, the physics of the problem, not the geometry, governs the nature of the mesh and the resulting solution for the fields.

Even though the thermal load in Figure 16 is exactly 100 times greater than that of Figure 14, the resulting temperatures are not 100 times greater, because the convection boundary condition is much more effective in cooling the slab. This follows because the convection thermal gradient is proportional to the difference between the ambient and actual temperatures. In Figures 13 and 15, however, $a = 20$, which limits the convective flux to smaller values.

b. Analysis of Nonlinear Coupled CEM and TM Problem. The coupled problem will be solved on the axisymmetric grid shown in Figure 17. The numbers to the left and below the figure label the rows and columns, whereas the other numbers label the nodes of the grid.

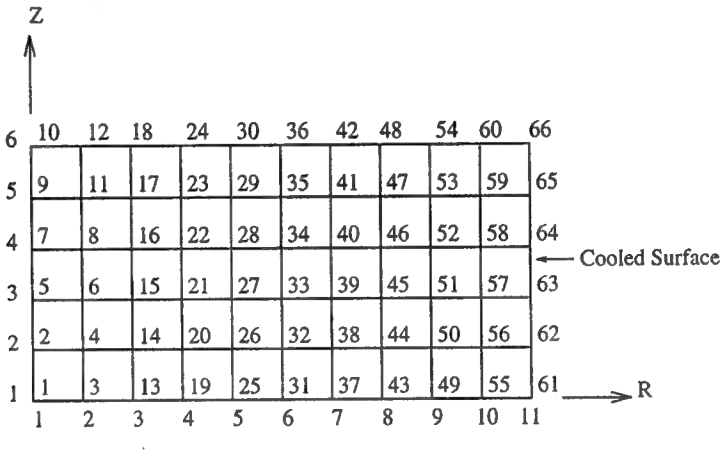


Figure 17: An axi-symmetric grid for the coupled EM-TM problem.

In applying the algorithm of Figure 10, we start with the trial temperature of 20° C, which from Figure 7 produces the following interpolated conductivity data:

$$\begin{aligned} \sigma &= 1.67 \quad \text{for } E = 0 \\ &= 1.85 \quad \text{for } E = 2200 \\ &= 2.3 \quad \text{for } E = 4400 \\ &= 2.72 \quad \text{for } E = 6600 \\ &= 3.2 \quad \text{for } E = 8800 \end{aligned}$$

The first two steps of the algorithm are illustrated next:

Step I: Solve $\nabla \cdot (\sigma \nabla V) = 0$ (A nonlinear EM problem)

$\sigma = \sigma(E, T)$ at $T = T_0 = 20°$ C

Let $T_i = 20°$ C, $T_a = 20°$ C, and $a = 20$, $V_0 = 1$ Volt

Result is $E = 6.6814 \times 10^2$ V/m, uniformally distributed throughout the material

$\sigma(T = 20°, E = 668.14) = 1.72$ (interpolated from graph)

Step II: Solve $\nabla \cdot (\lambda \nabla T) = \sigma(\nabla V)^2$ (A linear TM problem)

$T_i = 20°$ C, $T_a = 20°$ C, $a = 20$, $\lambda = 4.5$ (independent of T or E)

thermal load: $\sigma(\nabla V)^2 = 1.72 \times (668.14)^2 = 7.68 \times 10^5$ Watts/m^3

In order to proceed into the third step, which is another electrical problem, we must decompose the original body into ten parts, in order to accomodate TOPAZ3D. Each part corresponds to one of the columns of the grid. The third step and its results are listed next:

Step III: Solve $\nabla \cdot (\sigma \nabla V) = 0$ (A nonlinear EM problem)

$\sigma = \sigma(E, T(r)) = \sigma(E, r)$ (electrically inhomogeneous body because of temperature variation)

rerun problem with original ten sections of homogeneous (but nonlinear) materials

the results listed below are the input to a fitting program that uses DT_NURBS to interpolate the conductivity data

Results:

Node No.	Temp	E(V/m)	Interp. σ	vol. flux density=σE^2
1	43.49	816.01	2.118	816.01*816.01*2.118
3	43.44	815.82	2.117	815.82*815.82*2.117
13	43.31	815.21	2.115	815.21*815.21*2.115
19	43.09	814.22	2.111	814.22*814.22*2.111
25	42.79	812.62	2.106	812.62*812.62*2.106
31	42.41	810.42	2.099	810.42*810.42*2.099
37	41.93	807.62	2.091	807.62*807.62*2.091
43	41.38	804.42	2.082	804.42*804.42*2.082
49	40.73	800.62	2.071	800.62*800.62*2.071
55	40.01	796.22	2.059	796.22*796.72*2.059
61	39.20	794.02	2.046	794.02*794.02*2.046

The thermal volume flux density shown in the last column is then input into the next thermal problem, and that result gives the input to another electrical problem. The result after that cycle and the next few iterations follows:

Node No.	Temp	E(V/m)	Temp	E(V/m)	Temp	E(V/m)	Temp
1	44.665	823.22	46.197	833.215	45.761	830.415	46.034
3	44.560	823.02	46.090	832.815	45.649	830.015	45.924
13	44.316	822.02	45.839	831.616	45.389	828.815	45.667
19	43.920	820.02	45.433	829.616	44.967	826.615	45.250
25	43.370	817.02	44.869	826.616	44.382	823.416	44.672
31	42.667	813.02	44.151	822.416	43.637	819.215	43.935
37	41.812	808.22	43.279	817.415	42.734	814.015	43.043
43	40.807	802.42	42.259	811.615	41.677	807.815	41.991
49	39.655	795.62	41.096	804.615	40.472	800.619	40.808
55	38.358	787.82	39.794	796.815	39.123	792.615	39.477
61	36.922	783.62	38.360	792.819	37.638	788.415	38.011

c. Another Analysis of the Nonlinear Coupled CEM and TM Problem. We are going to revisit the last problem, but with a new conductivity profile, in which the values shown in Figure 7

are all multiplied by 10. This transforms the material into a 'super' Barium Titanate. Furthermore, we raised the value of the convection coefficient, a, to be 200. Our interest is in seeing if there are significant changes in the way the algorithm of Figure 10 must be applied, or if there are significant changes in the solution for the electric and temperature fields.

When we applied the algorithm in the same manner as before, we found that the convergence was unaffected, and that the results, as shown in Table 4, were quite reasonable. The first row associated with each node corresponds to the temperature at that node, and the second row the electric field.

Table 4: Results of Iterative Solution Algorithm.

	Iteration No.							
Node No.	1	2	3	4	5	···	21	22
1	23.2023	27.8183	23.6086	26.9054	23.9475	···	24.9543	25.1567
	68.80	71.72	69.08	71.16	69.28	···	70.08	69.92
3	23.1951	27.8122	23.6018	26.8993	23.9408	···	24.9479	25.1503
	68.80	71.72	69.08	71.16	69.28	···	70.08	69.92
13	23.1782	27.7981	23.5858	26.8850	23.9252	···	24.9330	25.1355
	68.80	71.72	69.06	71.14	69.28	···	70.06	69.92
19	23.1508	27.7751	23.5597	26.8617	23.8998	···	24.9087	25.1114
	68.80	71.70	69.04	71.12	69.26	···	70.04	69.92
25	23.1127	27.7431	23.5235	26.8294	23.8644	···	24.8750	25.0778
	68.78	71.68	69.02	71.12	69.24	···	70.02	69.88
31	23.0638	27.7021	23.4771	26.7879	23.8191	···	24.8317	25.0348
	68.74	71.66	69.00	71.10	69.22	···	70.00	69.86
37	23.0041	27.6521	23.4204	26.7373	23.7638	···	24.7789	24.9823
	68.70	71.64	68.98	71.06	69.18	···	69.98	69.82
43	22.9337	27.5631	23.3535	26.6776	23.6985	···	24.7165	24.9203
	68.66	71.60	68.94	71.02	69.14	···	69.94	69.78
49	22.8526	27.5250	23.2765	26.6088	23.6232	···	24.6447	24.8489
	68.62	71.56	68.90	70.98	69.10	···	69.90	69.74
55	22.7608	27.4480	23.1892	26.5309	23.5381	···	24.5634	24.7681
	68.56	71.52	68.84	70.94	69.06	···	69.86	69.70
61	22.6583	27.3621	23.0918	26.4440	23.4430	···	24.4726	24.6779
	68.52	71.48	68.80	70.92	69.04	···	69.84	69.68

Hence, we conclude that the algorithm performs stably over a wide range of material and physical parameters for this problem.

It is clear from these two examples that, though the algorithm converges, it does so slowly. To be sure, the results could be useful for some engineering design purposes after only two or three iterations, but it would be attractive to be able to estimate the limit of the sequence of iterations.

There are a number of sequence-accelerating algorithms that exist in the mathematics literature, and one that we have applied to this problem is called the θ-algorithm [47], which is defined by:

$$\theta_{2k+2}^{(n)} = \theta_{2k}^{(n+1)} + \frac{\left[\theta_{2k}^{(n+2)} - \theta_{2k}^{(n+1)}\right]\left[\theta_{2k+1}^{(n+2)} - \theta_{2k+1}^{(n+1)}\right]}{\theta_{2k+1}^{(n+2)} - 2\theta_{2k+1}^{(n+1)} + \theta_{2k+1}^{(n)}}$$

$$\theta_{2k+1}^{(n)} = \theta_{2k-1}^{(n+1)} + \frac{1}{\theta_{2k}^{(n+1)} - \theta_{2k}^{(n)}}$$

$$\theta_{-1}^{(n)} = 0$$

$$\theta_{0}^{(n)} = S_n$$

In Table 5 we include the results of applying it to the iterations shown above for Node No. 1:

Table 5: Result of θ-Algorithm Iterations.

n	$\theta_0^{(n)}$	$\theta_2^{(n)}$	$\theta_4^{(n)}$	$\theta_6^{(n)}$	$\theta_8^{(n)}$	$\theta_{10}^{(n)}$	$\theta_{12}^{(n)}$	$\theta_{14}^{(n)}$
1	23.2023	25.5301	25.1896	46.7976	25.2572	25.0773	25.0772	25.0785
2	27.8183	25.3972	26.3146	25.5808	25.0772	25.0773	25.0775	
3	23.6086	25.3042	25.1399	24.9870	25.0773	25.0778	24.8970	
4	26.9054	25.2375	24.7454	25.0774	25.0772	25.1448	25.0262	
5	23.9475	25.1883	25.0401	25.0772	24.9579	25.0097		
6	26.3346	25.1534	25.0753	25.0750	25.0154	25.0789		
7	24.2122	25.1282	25.0785	25.0218	25.0510	25.0604		
8	25.9539	25.1088	25.0601	25.0787	25.0801			
9	24.4269	25.0951	25.0741	25.0499	25.0694			
10	25.6894	25.0859	25.0794	25.0693	25.0638			
11	24.5870	25.0797	25.0700	25.0651				
12	25.5126	25.0759	25.0685	25.0638				
13	24.7083	25.0721	25.0643	25.0638				
14	25.3907	25.0678	25.0638					
15	24.8017	25.0649	25.0638					
16	25.2943	25.0638	25.0638					
17	24.8690	25.0638						
18	25.2291	25.0627						
19	24.9231	25.0618						
20	25.1837							
21	24.9543							
22	25.1567							

Note that the initial data, $\theta_0^{(n)}$, which are the results of the TOPAZ iterations, still differ by 0.2024 after twenty-two iterations. After fourteen stages of application of the theta algorithm, however, the difference between $\theta_{14}^{(0)}$ and $\theta_{12}^{(0)}$ in the top row[1] is only 0.13×10^{-2}, which means that we have reduced the error by more than two orders-of-magnitude. Because the theta algorithm is quite fast, we can recommend it for application to iterations of this sort. This algorithm has been applied to the computation of Green's function in electromagnetic problems [48].

d. A final analysis of the nonlinear coupled problem. In this final study of Canonical Problem No. 2, we attempt to force a condition that might require different grids to be used for the electrical and thermal analyses. That is, we seek conditions on the various parameters of

[1] Convergence of the θ-algorithm is along this row.

the problem that would cause the gradient of one field to greatly exceed that of the other. Our approach is purely empirical, but it is guided by the guess that we will need to operate in that range of Figure 7 for which conductivity changes rapidly with temperature. This is above the knee, which is roughly 100°C. In this region of operation, a small change of temperature ought to produce a large change in electric field.

Our results are summarized as:

(a) In the range $0°C < T < 100°C$, for example, when T is about 60 °C and E is about 1170 V/m, the largest change of temperature between two adjacent nodes is 2.7% and the largest change of electrical field is about 0.7%.

Node	$E(V/m)$	$T(°C)$
1	1181.22	64.3650
3	1180.82	64.2576
13	1179.22	64.0070
19	1176.62	63.5988
25	1173.22	63.0311
31	1168.82	62.3031
37	1163.22	61.4145
43	1156.43	60.3652
49	1147.82	59.1550
55	1138.22	57.7839
61	1133.62	56.2519

There is no need to change the grid.

(b) In the range $100°C < T < 140°C$, for example, when T is about 120 °C and E is about a few hundred V/m, the largest change of temperature between two adjacent nodes is 3.4%, and the largest change of electrical field is about 20%.

Node	$E(V/m)$	$T(°C)$
1	340.863	127.087
3	345.608	126.828
13	361.463	126.223
19	390.711	125.238
25	434.027	123.867
31	491.436	122.110
37	563.231	119.965
43	665.433	117.432
49	800.233	114.511
55	954.233	111.202
61	1036.05	107.504

In this case, it may be necessary to use two different grids to solve the electrical problem and the thermal problem.

Now we consider the key parameters that may force one to use different grids. According to the results that are tabulated above, we know that when the temperature is above 100°C, a slight change of temperature will cause a large change of electric field, and this is depends solely on the non-linearity of the material. One must consider several factors in order to get the temperature above 100°C for this particular material, but the thermal load $(\sigma(E,T) * E^2)$ is the most important

one. The ambient temperature, initial temperature, and convection coefficient also play key roles in this problem.

For instance:

(a) When the thermal load is about 3.0×10^6 to 4.0×10^6, we need to consider using different grids. It doesn't matter what the ambient temperature is as long as the convection coefficient is small enough to keep the temperature in the range between 100°C and 140°C. For this particular type of material, if we rescale the electrical conductivity σ by a factor of 1.67, the thermal load will then be in the range of 3.0×10^6 to 4.0×10^6, and the results shown in the second table are obtained (a relatively big change in E-field and small change in T).

(b) If the ambient temperature is in the region of 100°C to 140°C, and if the convection coefficient is large enough to keep the temperature within the body close to the ambient temperature, then it may be necessary to use different grids for the T and E calculation. Ambient temperatures, however, are normally around 20°C, so that this case is rare.

(c) In conclusion, therefore, the question of whether or not to change the grid for the E and T calculation depends upon the non-linearity of the material for a problem with a simple geometry, such as Canonical Problem No. 2. For Canonical Problem No. 3, however, the situation is very different and further studies should be done.

Canonical Problem No. 3

a. **Analysis of uncoupled CEM and TM problems.** As before, we gain some insight into the nature of the coupled problems by first running uncoupled problems, except that we take the thermal loading to be $\sigma(\nabla V)^2$, where V is computed from the pure electric problem. The upper electrode has a radius of 3mm.

Electrical problem. Let $V_0 = 1$ V in Figure 8; then the potential distribution and electric field are shown in Figures 18 and 19. We attempt to produce a more accurate solution by putting more

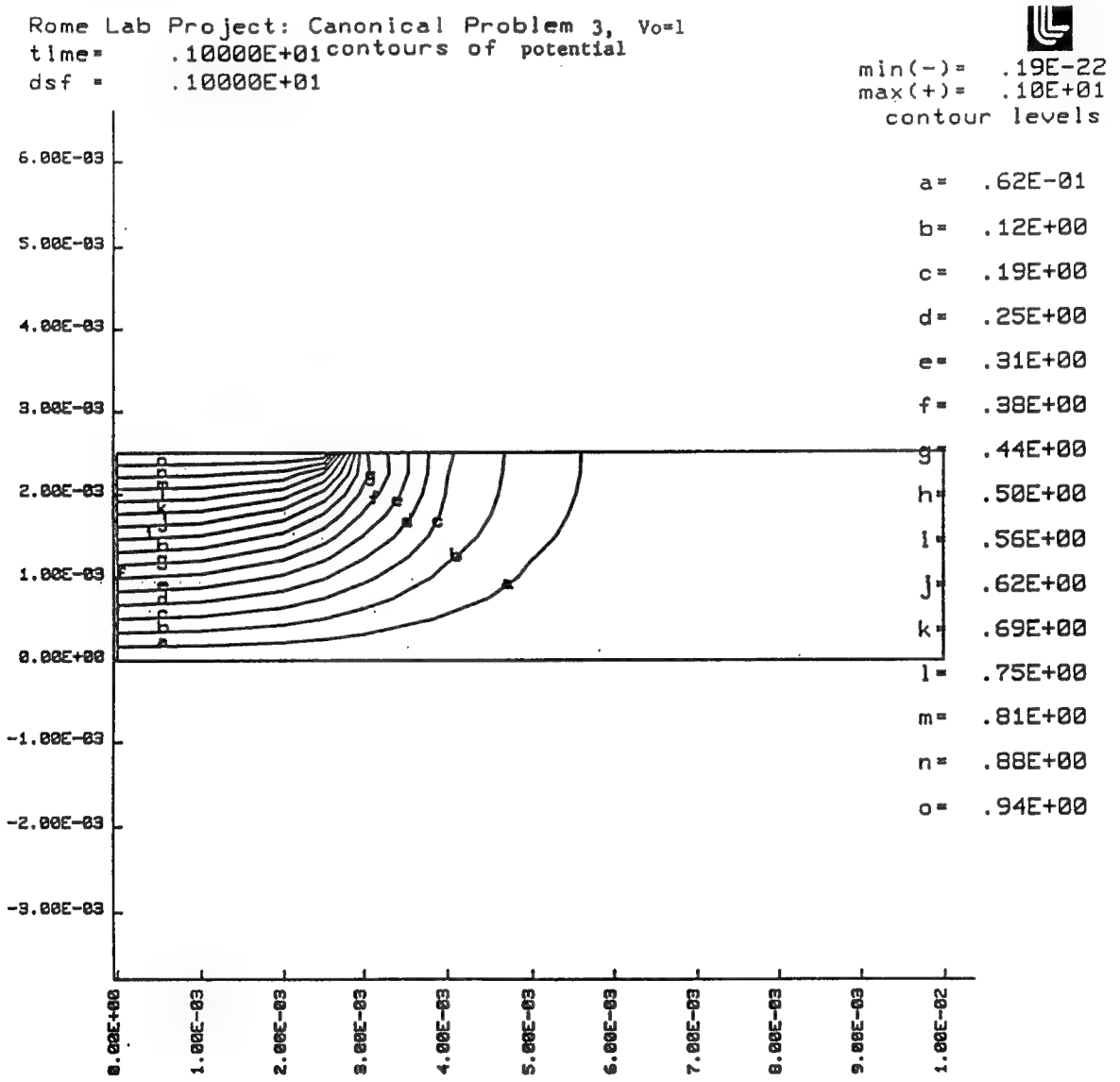


Figure 18: Potential distribution for Canonical Problem No. 3, when the electrical and thermal problems are uncoupled.

radial cells in the vicinity of the upper electrode; the result is shown in Figure 20. One must generally resort to special techniques, such as the use of adaptive optimal gridding, based upon *a posteriori* error estimates, or the use of special basis functions in order to accurately reproduce the electric field in the vicinity of the edge of the upper electrode.

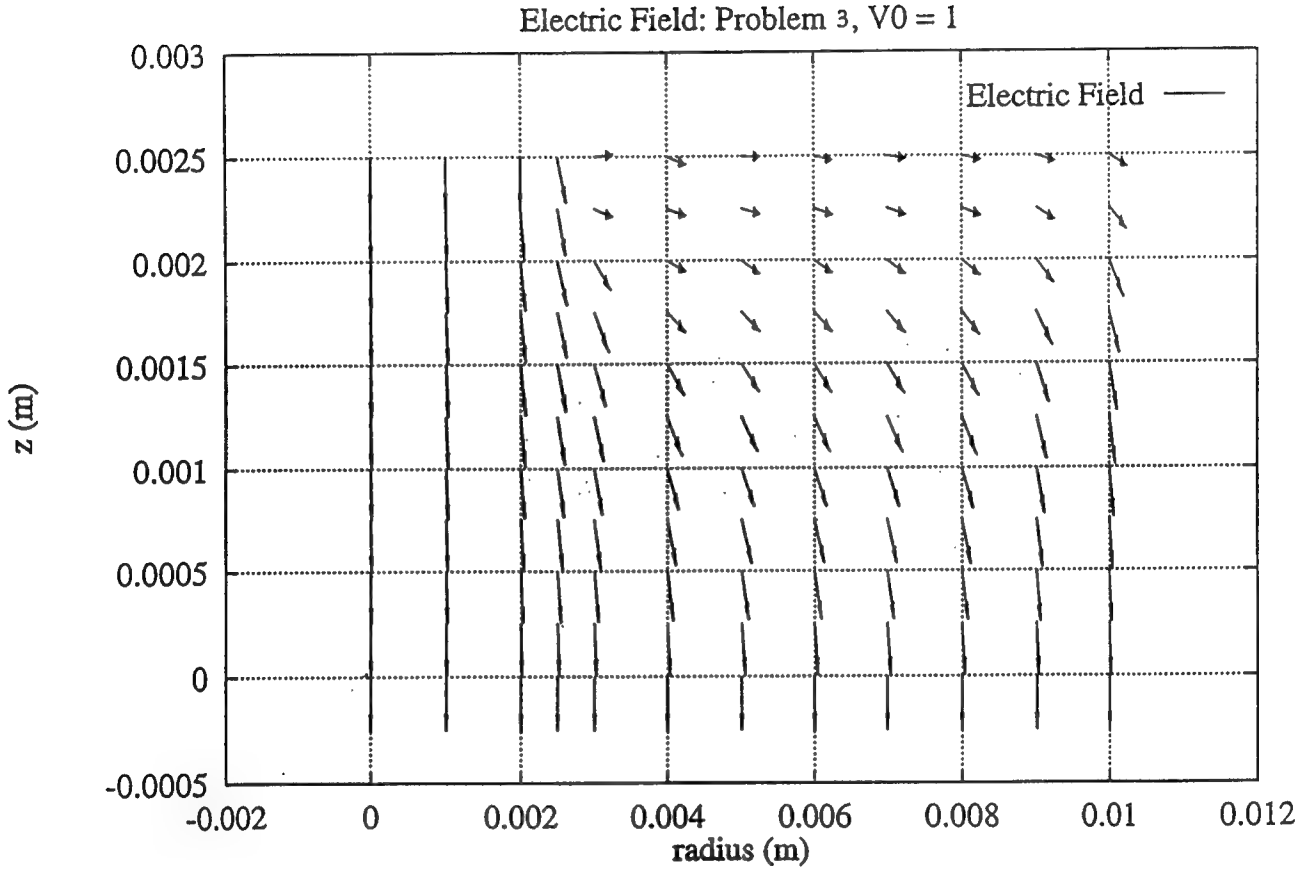


Figure 19: Electric field distribution for Canonical Problem No. 3, when the electrical and thermal problems are uncoupled.

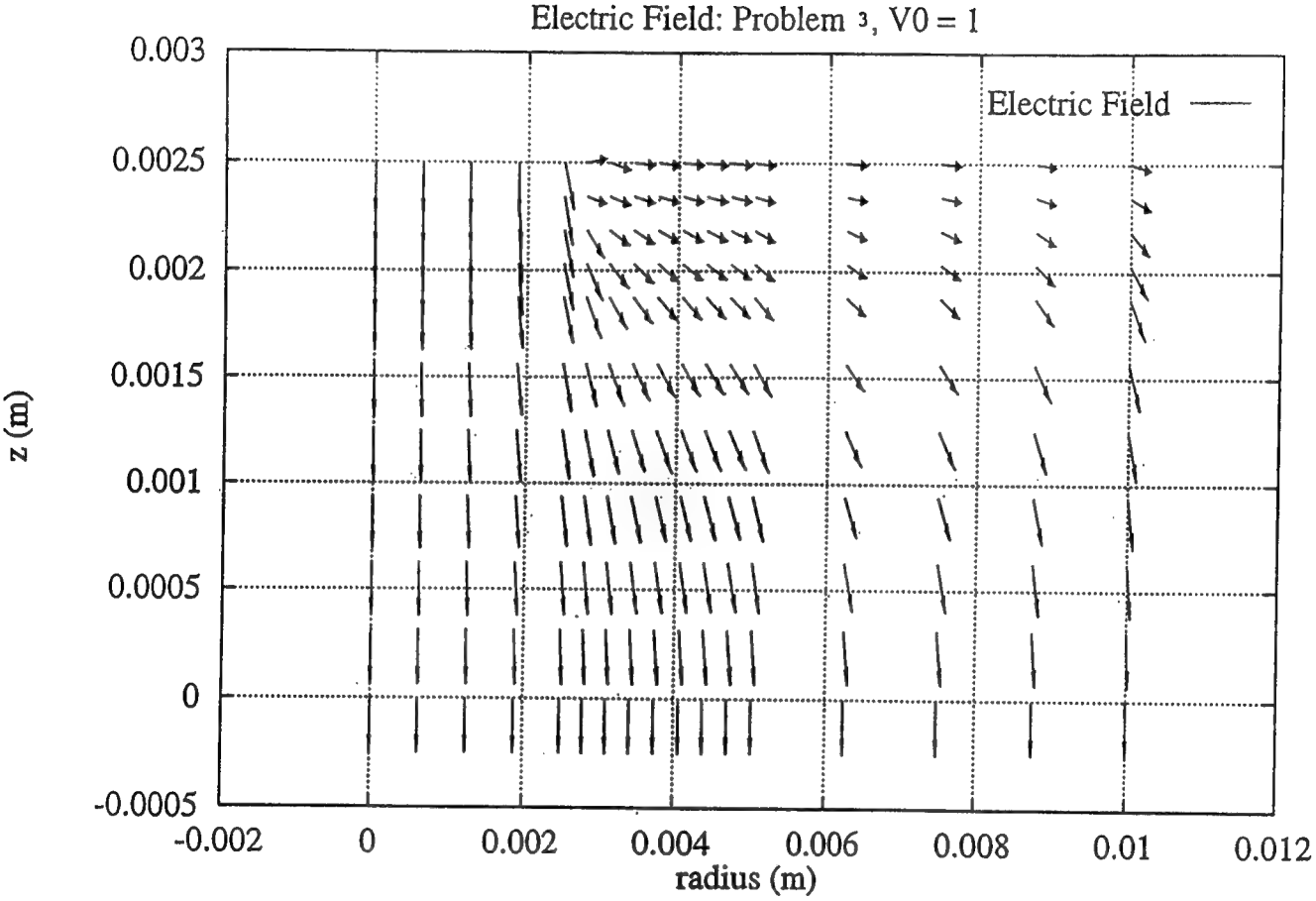


Figure 20: Electric field distribution for Canonical Problem No. 3, when the electrical and thermal problems are uncoupled. A slightly more refined grid, compared to that of Figure 19, is used.

Thermal problem. The thermal problem was solved using the same mesh of Figures 18 and 19. The temperature profile for the case in which $V_0 = 1V$, $a = 20$ is shown in Figure 21, and for $V_0 = 10V$, $a = 20$ in Figure 22. Clearly, even though the temperature profile in Figure 21 is not

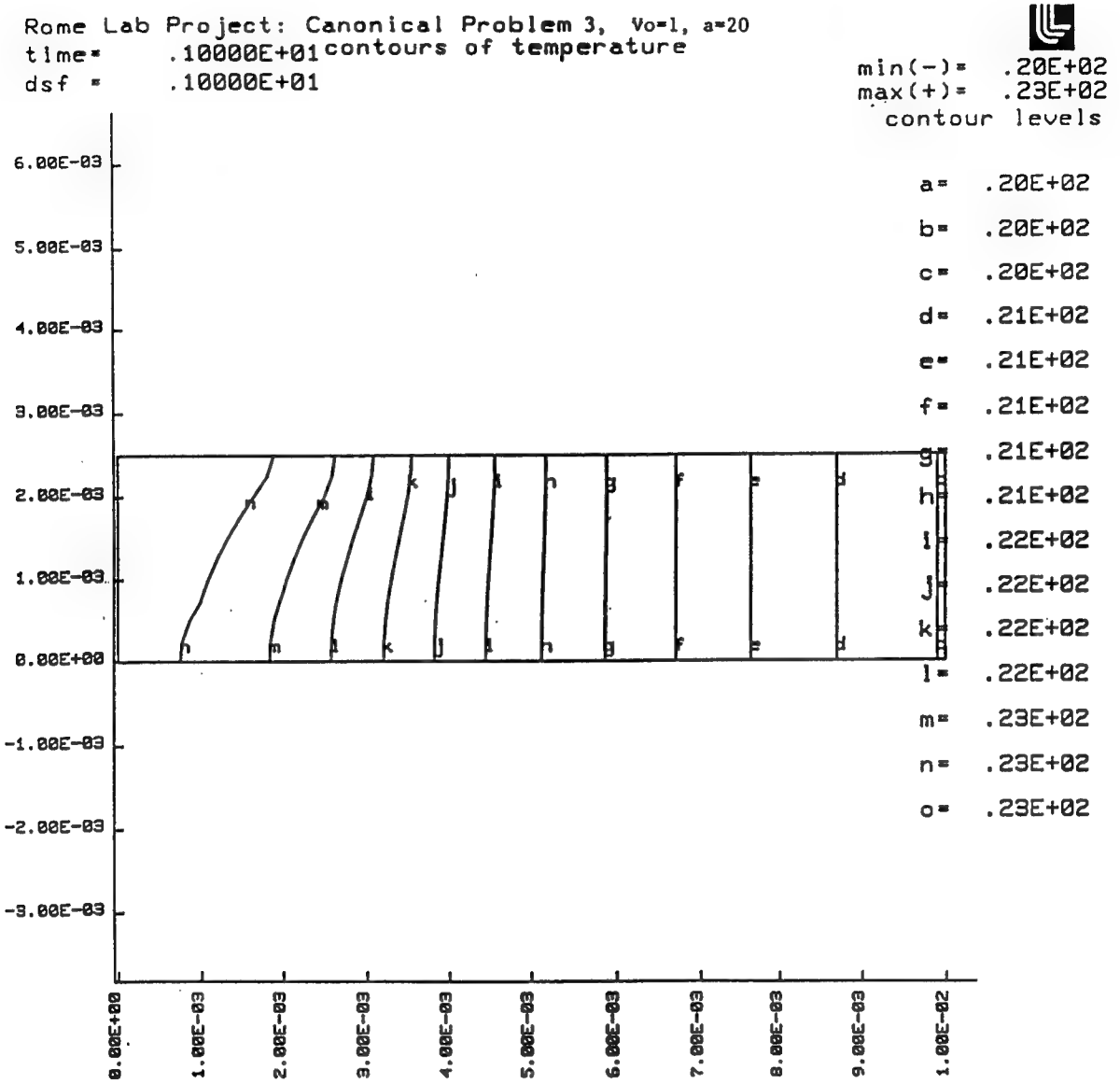


Figure 21: Isothermal contours when $V_0 = 1V$, $a = 20$ for Canonical Problem No. 3. The right-hand edge is the exposed surface at which convection takes place. These parameters produce a modest thermal gradient.

uniform, the radial gradient is modest, in contrast to the result of Figure 22, which shows a much greater gradient. This result is dependent entirely upon the excitation voltage, not the geometry, which is the same in both problems.

b. Analysis of coupled CEM and TM problems. The solution of this coupled problem is obtained in precisely the same way as Problem No. 2, the only difference being that the initial body must be decomposed into N_e bodies, where N_e is the number of elements in the thermal mesh. This is due to the fact that the electric field is no longer relatively uniform with z, as it was

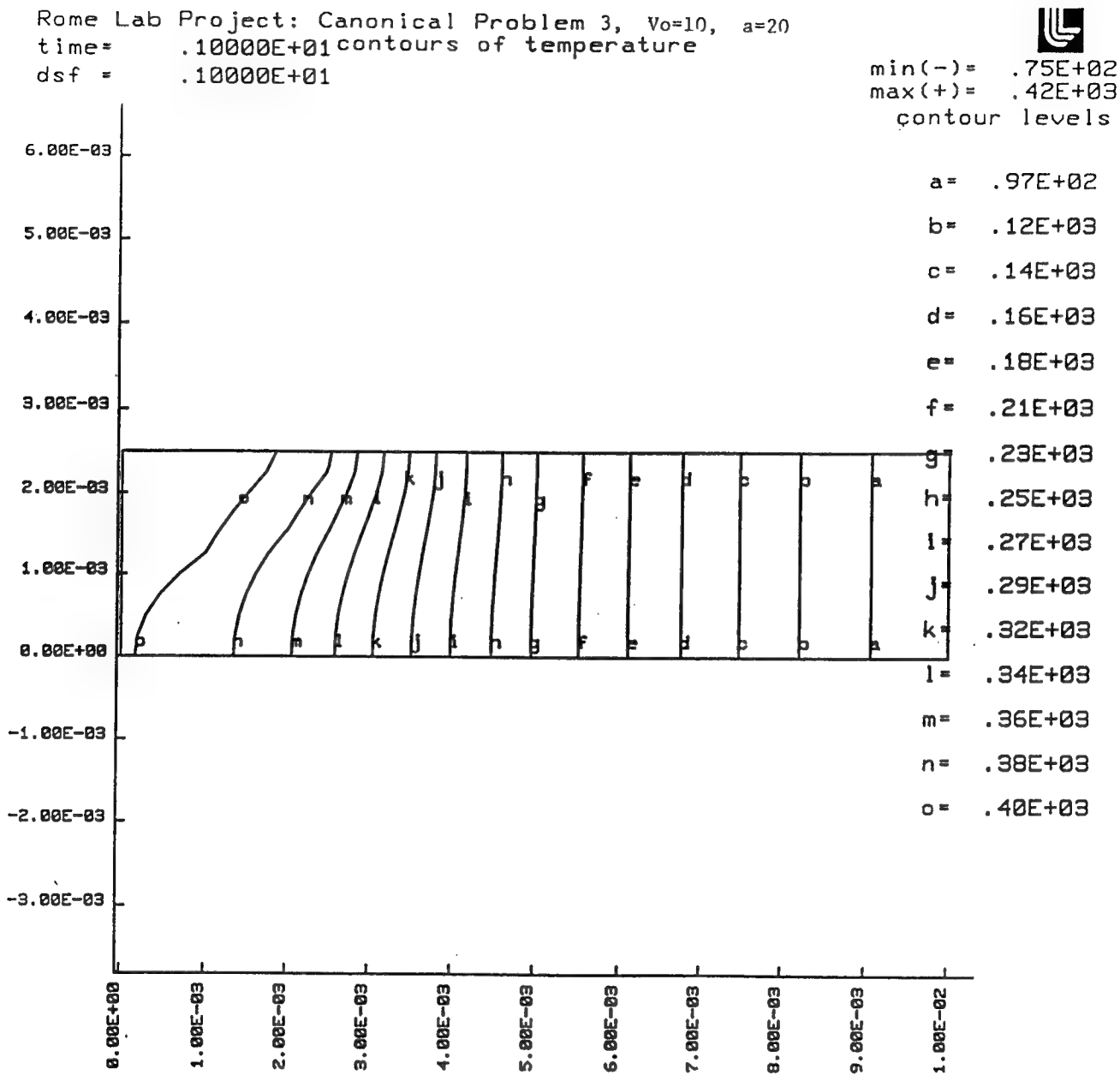


Figure 22: Isothermal contours when $V_0 = 10V$, $a = 20$ for Canonical Problem No. 3. The right-hand edge is the exposed surface at which convection takes place. These parameters produce a significant thermal gradient.

in Problem No. 2. This has the disadvantage of requiring more preprocessing of data, in order to continue the iterations. Of course, this would be done with a 'supervisor' in a production code.

The grid that is used to solve the coupled problem is shown in Figure 23. It is identical to that used for Canonical Problem No. 2, except that the upper electrode extends only to node 18, and nodes 10 and 11 are interchanged. The slab is 2.5mm thick, and its radius is 10mm. The upper

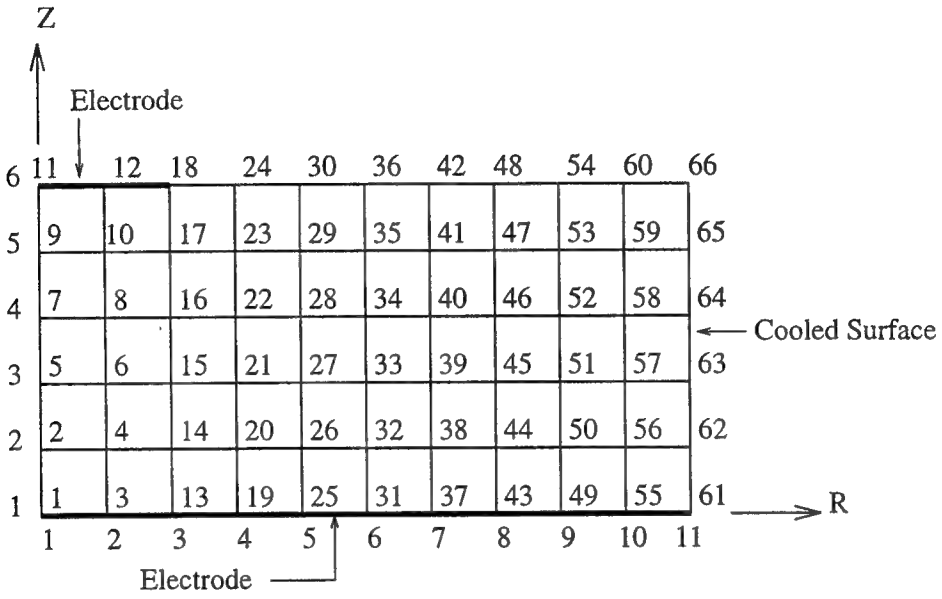


Figure 23: Mesh for Canonical Problem No. 3.

electrode has a radius of 2mm, in contrast to the situation in Paragraph 4.1, for the uncoupled problem.

The only calculation that we will report for the coupled problem concerns the application of the theta-algorithm for the temperature of Node 1. The results are shown in Table 6.

Table 6: Result of θ-Algorithm Iterations for Node 1.

n	$\theta_0^{(n)}$	$\theta_2^{(n)}$	$\theta_4^{(n)}$
1	27.3913	24.5809	24.6234
2	22.3278	24.4499	
3	26.6362	24.3397	
4	22.4491	24.2467	
5	26.0754		
6	22.5634		
7	25.6480		

Note that after only four iterations, the θ-algorithm has reduced the estimated error by two orders-of-magnitude.

In order to check the convergence of the solution with mesh fineness, we doubled the number of cells in the grid, and computed the change in the electric field at nodes 18 and 17 in the original grid. (There is no coupling between thermal and electrical variables in this test.) The electric field

at node 18, which is at the edge of the upper electrode, increased by 35%, whereas the field at node 17 decreased by only 7.4%. The field values at nodes 1, 2, 5, 7, 9, and 11, i.e., along the axis of symmetry, changed by amounts varying between 1.6% and -5.8%. This indicates that the field varies smoothly in space, except at the edge of the upper electrode, as we know from theoretical considerations.

The optimum grid for this problem would have enough cells around the edge of the upper electrode, such that the error distribution throughout the entire grid is uniform. This matter is discussed in more detail in Chapter 4. We can summarize this point in another way by comparison with the mesh required for Canonical Problem No. 2. The geometry of that problem is identical to that for this one, except that the boundary conditions are significantly different, and this induces considerably different mesh requirements.

4 RELATED MATTERS

The driving issue of this research has been associated with computational matters for coupled problems, especially those relating to the reusability of grids and grid-related data. Up to this point, however, we have not spent a great deal of time worrying about the structure or nature of the grid, especially about refining it in order to improve the accuracy of the solution. Indeed, that was not a part of the research aims. The grid is, of course, crucial to the solution of any discretized functional equation, especially so when the discretization involves the finite-element method. A poor grid may not even allow the analysis modules to work. Thus, in this chapter we turn our attention to a number of related matters, especially those involving the grid, and use this chapter, together with Chapter 3, as a jumping-off point for proposing future work, as described in Chapter 5.

The first part of this chapter consists of a summary and literature search that deal with a number of things, such as the display of scalars and vectors, to a review of hardware and software, to the very important subject of mesh generation and error analysis. These topics are of fundamental importance in relating computational electromagnetics to computer-aided engineering and design. The second part of the chapter deals with the subject of extrapolation of solutions to reduce the effects of truncation error caused by meshing.

4.1 CEM Computational Models and Mesh Generation

Several sophisticated numerical algorithms are introduced in [9]: adaptive meshing, multigrid, and semi-implicit schemes. Adaptive techniques require eleborate and complicated data structures and must attempt to minimize the computational overhead of the error estimation and of the implementation of the adaptive process. The adaptive methods consist of refining the mesh size h (h-method), increasing the density of grid points by relocating nodes (r-method), and increasing the order, p, of the interpolating polynomial (p-method), or combinations of these techniques. Adaptive local grid refinement techniques, both fixed and dynamic, afford enormous potential for local accuracy improvements in many large-scale problems.

Multigrid methods, or multilevel methods, are a class of iterative algorithms designed to improve the convervgence rate of relaxation methods for solving a set of linear equations $Ax = b$. Conventional relaxation methods, such as Jacobi, Gauss-Seidel, and SOR, are more effective at eliminating error components at certain frequencies of the spectrum of the grid operator. Multigrid methods try to improve the convergence rate by constructing a grid operator that eliminates those frequencies which the relaxation method does not handle well. Multigrid methods arose from applications posing an elliptic PDE on a spatial domain, where discretization of the PDE resulted in a linear set of equations. The domain of the set of equations has direct physical relevance: it represents the underlying physical grid. Thus, the term *geometric* multigrid[2].

Multigrid algorithms solve problems by starting with a fine mesh, then sequentially coarsening it through several levels, and then returning to the fine mesh. On each mesh a few "passes" or iterations are used to get approximate solutions, which are then interpolated or restricted to the next mesh. Multigrid algorithms process a cycle from the fine to the coarse grids and back to the fine grids, but on each grid level the problem can be treated in parallel similarly to the parallel algorithms for "simple" grids. Parallel grid algorithms are usually iterative methods that calculate the value of a grid function at one point as a function of values defined at neighboring points (also called relaxation). The iteration can be characterized as Jacobi-type, wherein the new iterate at

[2]personal communication from Chris Doyle

a grid point is calculated by using only old neighboring values, or Gauss-Seidel-type, which use already calculated new neighboring values. Jacobi-type methods are completely parallel since the calculation in each grid point can be performed independently. If the number of grid points is N, then the parallelism is also N. The parallelism of Gauss-Seidel methods depends on the order in which the grid points are processed.

High speed computers, with large memories, together with improved numerical techniques, have led to the solution of ever more complex and higher-dimensional problems. Because of this, computer graphics has come to play an important part as a post-processor. In order to address increasingly larger volumes of output data, and because of the difficulty of grasping solutions expressed as multidimensional fields, we require highly efficient and simple means of visualizing the data. Computer graphics using suitable shapes and colors (hue and lightness) can assist visulization of output data, and interactive computer graphics makes the best use of human perception.

Paper [10] discusses the display of distributed scalars and vectors, especially electric and magnetic fields. As an example of distributed multi-scalars, one may display the relationship between the distribution of eddy-currents caused by a magnetic field and the magnetic flux density distribution by displaying the eddy-current by arrow length, and the magnetic flux density by the color of each arrow. A stereograph is very useful for displaying vectors distributed in a three-dimensional field. Arrows and stream lines with pseudo-colors are useful. For example, pseudo-colors on the flux lines give the value of magnetic flux density, and the tangent at each point of the line gives the direction of the flux line. One can also use graphical techniques to display such things as errors due to the choice of the unknown variable, the approximation functions, and on the display techniques, themselves.

The authors of [11] present an analysis of the present and prospective generation of computer hardware and system software for computer-intensive applications of computer-aided analysis and design. They conclude that of the three classes of machines: mainframes and superminicomputers; supercomputers and minisupercomputers; and workstations, superworkstations, and personal computers, the latter will assume the dominant role in computer-aided design and analysis. This is due to the intrinsic matching with an interactive environment, the improvements in performance already available and coming (this was written long before the announced 100 MIPS workstations, which are to be out by the end of this year), and the very significant decrease in prices and price/performance ratios. Of course, these machines are possible because of the availability of faster CISC and/or RISC processors, and because they are designed with Cray-like vector features and low-level parallelism.

A CAD user should be concerned with two things above all: a careful benchmarking of his machine, to evaluate realistic performances, and the quality of the system software. The authors of [11] suggest that the Livermore Kernels, rather than the LINPACK benchmark, is the most representative environment of an electromagnetic analysis code. This benchmark is built using a set of the most computationally intensive kernels of scientific programs in use at the Lawrence Livermore National Lab. The authors further conclude that manufacturers of hardware will consider the key feature in selling their products to be such software-related matters as operating systems, the compilers, and utilities, such as graphics library/device-drivers sets.

Finally, massive parallelism will be applied to solve specific analysis problems, with massively parallel systems becoming more readily available.

Adaptive meshing is an active field of research in computational electromagnetics, just as it is in computational fluid dynamics, and computational mechanics. It is motivated by the need to produce more robust and user-friendly finite-element analysis environments. For CEM analysis, the range of differential equations and material properties of interest in practical applications is very wide, so that proposed algorithms should be as robust, reliable and flexible as possible if they

are to be used in general purpose electric and magnetic analysis codes. An additional difficulty for these specific areas is the very significant geometrical complexity of a large number of devices of practical interest. This poses additional challenges to adaptive meshing algorithms, that should be able to cope automatically with intricate interface boundaries, that separate regions that often contain very different material properties, including nonlinear materials. The remaining papers deal with these, and other issues.

Paper [12] describes a 3D automatic mesh generator that is used in the Laplace-solver, Phi3d. Phi3d uses the boundary-integral method, wherein potentials and fields are computed from fictitious source densities uniquely situated on the surfaces of homogeneous regions of space. Thus, only the surfaces need to be discretized, and the mesh generator that does this discretization is based on a combination of two different main algorithms:

- automatic generation of triangles for plane surfaces, as is usually done in 2D, including geometric improvements and refinements,
- automatic generation of quadrangles for curved surfaces, using linear Coons patches.

Furthermore, some simple surfaces are directly meshed when generated from a data base of primitive objects, such as spheres, cubes, cylinders, etc., by translating such a primitive surface along a line, or by rotation around an axis.

Paper [13] deals with the automatic generation of 3D meshes with a prescribed boundary. The algorithm generates a tetrahedral mesh using a variant of the Delaunay-Voronoi technique for two-dimensions. A quality factor for each element of the mesh is introduced, which is a measure of the ratio of the radius of the largest inscribed sphere within the tetrahedron, to the longest edge of the tetrahedron. This quality factor should be close to unity for a mesh of good quality.

A new method for generating tetrahedra for open-boundary, 3D, finite-element problems in magnetic field analysis is presented in [14]. A topological mapping is used for the multi-media region, which includes cores and coils, and a multi-layer space-dividing technique for the surrounding air. Typically, the interior region, which contains different media, requires a dense, complex, mesh, whereas the mesh in the exterior region is usually much simpler, and uses coarser elements near the ultimate boundary.

The aim of [15] is to discuss the sources of error and their influence on the different field quantities of a finite-element calculation, and to show some methods to increase and control the accuracy of the solution. A finite-element solution that is based on the use of scalar and vector potentials may have a number of error sources:

- The **approximation** of the potential causes an error according to the order of the shape functions used.
- The finite-element approximation of the field does not generally satisfy the condition $\nabla \cdot \mathbf{B} = 0$, when the scalar potential is used, or $\nabla \times \mathbf{H} = 0$ when the vector potential is used. In the second case, due to this **boundary error**, the tangential component of the magnetic field strength, $\mathbf{H}$, is not continuous on the boundary of two adjacent elements.
- Due to the **discretization** of the problem area by means of finite elements, large errors may occur if the mesh is either too coarse, or unsymmetrical.
- There exists an error related to the **residual** left over after the solution of the system of equations. This error depends upon the computer precision and the method used for the solution of the equation.

- In **post-processing**, approximations are used to evaluate the potentials and to derive local (magnetic induction and field strength) and integral (magnetic energy, flux, and forces) field-quantities. These approximations can also be sources of error, as, for example, in calculating forces by means of the Maxwell stress-tensor.

- Besides these basic errors of the solution method, errors may be introduced also in the **modeling process**: the location of the external boundary can affect the internal field if its distance to the internal field region is too small; saturation effects in iron requires a sufficiently detailed mesh in critical regions.

It is found that integral quantities, which are computed directly from the potential, such as flux and energy, are as accurate as the approximation of the potential. Local quantities, or integral quantities, such as induction, magnetomotive force, mechanical forces, and losses, which are derived from derivatives of the potential, i.e., from local fields are more sensitive to errors, not only in the potential, but also to discretization and boundary errors.

These errors can be reduced by using higher-order elements, or by using adaptive mesh refinement processes. A local adaptive refinement technique, restricted to sensitive regions, is developed in [15]. In this way accuracy of the local quantity is achieved with an effort that is less demanding than overall refinement of the mesh.

In [16] adaptive mesh refinements are incorporated into the boundary-integral method to produce accurate calculations of electromagnetic fields. The boundary-integral method requires a discretization of physical boundaries, only, and efficiently solves unbounded field problems. It is important to establish adaptive mesh refinement algorithms that are based on proper error estimates. This has been done for finite-elements to regulate local error fluctuations (indeed, that is the subject of many of these papers). Because the finite-element method is based on differential equations, its local error at any point can be easily estimated from the calculated results around the point. The error for the boundary-integral equation, however, is directly affected by all other calculated results along the entire boundary, and only a few results for adaptive mesh refinement have been reported so far. The authors of [16] introduce two adaptive mesh refinement schemes for the boundary-integral method, which allow the accurate calculation of electromagnetic fields in unbounded domains. Refinements are based on local error estimates of the calculated potential and its normal derivative on the boundaries. The simplest error estimates may be differences between two adjacent potential values and their normal derivatives. If either difference is greater than a prescribed tolerance, then a new node is added in the element containing the two adjacent noeds. The error at this node is calculated from the integral equation. These schemes are applied to the analysis of striplines with edge singularities.

The preceding paper used potential values at adjacent nodes as a local error estimator; in [17] a sensitivity analysis is developed that is based on the perturbation of local energy with nodal position, and this is used as the criterion for mesh refinement. This is a reasonable error criterion because:

- the sensitivity is higher in the region where the field is not sufficiently smooth, i.e., where the interpolation error is greater. The sensitivity is higher where the change of flux density with nodal displacement is considerably different in the elements surrounding the node under study, or in the deformed elements due to the displacement of this node. In the other case the sensitivity is zero if the changes in flux density vanish in the deformed elements, which is the case for linear potential distribution. It is evident that the sensitivity measures the change of the potential gradient instead of the potential;

- the sensitivity is higher in the region where the energy density is greater;

- the larger the element, the higher the sensitivity to nodal position.

The adaptive mesh generation algorithm of [17] is:

1. define the initial discretization points on the boundaries:
2. generate the mesh by Delaunay triangulation;
3. optimize the position of the interior nodes;
4. solve the problem;
5. calculate the sensitivity of the nodal position for all nodes;
6. determine the position of additional nodes;
7. insert successively the new nodes in the preceding mesh;
8. repeat steps 3 to 7 until the change of energy is lower than the given precision.

The results presented in [17] show that this algorithm consistently reduces the number of nodes for the final solution by a factor of two to three.

While adaptive refinement methods are well accepted in the solution of the Poisson equation of potential theory, little work seems to have been reported in solving the wave equation, or equations derived from it, such as the Helmholtz equation for nonzero frequencies. In [18] this question is addressed in the context of solving the scalar wave equation $\nabla^2\phi = k^2\phi$. Some significant differences exist between solving this equation using finite-elements and the Poisson equation. For example, the finite-element matrix equation takes the form:

$$A\mathbf{x} = \lambda T\mathbf{x},$$

where λ is an eigenvalue parameter. These eigenvalue correspond directly to the wavenumber, k, identified in the wave equation.

To apply adaptive mesh generation, whose twin goals are economy and confidence in solution, we may examine the error in the eigenvalue, λ, or the eigenvector, $\mathbf{x}$. A peculiarity of waveguide analysis is that we are usually not interested in all modes (eigenvalues) of a guide, but only in the dominant mode corresponding to the lowest eigenvalue, and sometimes also in the next for design reasons [18]. But the requirements of economy are different for any two modes. For example, a fine mesh close to the lower left corner of a rectangular guide may be required to compute the TM_{11} mode accurately, because that is where the field is concentrated, but such a mesh will not help in computing the TM_{51} mode, because the electric field components are virtually unchanged in that region [18]. Thus, we must identify the particular mode (or modes, as the case may be) that we seek, and apply the error criterion to the eigenvalue or eigenvector associated with it.

Paper [18] develops two adaptive schemes for eigenvalue problems. The first examines the change in the eigenvalue of the mode we are interested in, and repeatedly refines the mesh until the change is acceptable. This scheme is reliable, but not as econimical as the second. The second scheme examines the change in the normalized eigenvector from mesh cycle to mesh cycle, and accordingly refines the mesh at selective locations where the change is unacceptably high. This procedure accurately extracts a particular mode of the guide at great economy, as reported by the author.

Papers [19], [24], [31] present an analysis of both comparative and absolute performances of a set of adaptive strategies for mesh refinement in CEM. This analysis is performed by implementing, in

a uniform preprocessing environment, a number of adaptive meshing algorithms. These algorithms are used to solve a test case, that has an analytic solution, and features some of the most challenging situations for adaptive schemes, such as assessing both the visual quality of the resulting mesh and the difference between the real error of the solution and the estimate of the error provided by the methods.

The analysis, which uses triangular, first-order elements, is restricted to the use of *a posteriori* error estimators, that require a single problem solution and working out the error estimator on an element-by-element basis. These procedures are judged to be more reliable in implementation, since they minimize conceptual coding complexities in complicated geometries, when compared to procedures that are based on the estimate of a higher-order solution on the support area of each node. These procedures also provide a greater robustness in cases of practical interest with several different materials.

In particular, three methods have been identified as providing the most reliable and consistent results, namely the "local error problem" algorithm, the "complete residual" algorithm, and the "field difference" algorithm. The results of [19] are that the complete residual method, which is based on the estimate of error in the solution, paroduces meshes notably less thickened around sharp edges, and with fewer nodes in general. This provides acceptable overall meshes, but not meshes that are particularly optimized for the evaluation of local quantities. Much better results for the estimation of local values of gradient or curl of the solution have been obtained with the field difference procedure and with the local error problem approach. The latter procedure may be tuned, but at the cost of computational time.

In the mesh refinement algorithms developed in [19], [24], [31], the key quantity that determines which elements are to be refined is the refinement indicator η_i on the element Ω_i, and is defined by:

$$\eta_i = \sqrt{\kappa \frac{\|\nabla e_i\|^2}{\sum_{j=1}^{N} \|\nabla \tilde{u}_j\|^2} + (1-\kappa) \frac{\|e_i\|^2}{\sum_{j=1}^{N} \|\tilde{u}_j\|^2}}.$$

The quantity e_i is the estimated error on the finite-element solution $\tilde{u}_i$, N is the total number of elements in the whole domain Ω, and κ is a weighting factor to be selected in the range 0 to 1. This allows one to define η_i in terms of the estimated error on the solution, on its gradient (or curl), or on both. Another tuning parameter, $0 < \sigma < 1$, is defined by

$$\eta_i > \sigma \eta_{max},$$

where η_{max} is the maximum value of the refinement indicator among all elements. The parameters κ and σ are tuned during the various tests.

Papers [24, 31] apply the strategy of [19] to an electrostatic potential problem for an L-shaped two-dimensional region, having Dirichlet and Neumann boundary conditions. This problem has an analytical solution, which is compared to the finite-element solution.

Paper [20] returns to the subject of boundary-integral equations, and considers the application of knowledge-based methods for constructing meshes. Because the boundary-integral method solves for sources located on the boundaries between three-dimensional regions, it follows that a two-dimensional mesh generator, such as those used in finite-elements, is required.

The application of knowledge-based methods, such as expert systems, is effective in those problems which require experience and skills. In particular, when generating triangular meshes, there are many rules which must be applied when approximating unknowns, arranging boundary surfaces, considering the geometries of boundary surfaces and the boundary conditions of the computational model. Furthermore, know-how may be added to these rules, which will result in flexibility and

expandibility. In [20] a mesh generator using a knowledge base which consists of these rules is proposed. The knowledge base is separated from the execution module and described by LISP in order to achieve the flexibility and expandability. Furthermore, the kernel of the mesh generator is coded in LISP, because LISP is more flexible and expandable than procedural languages, such as FORTRAN and C. When the functions which are called according to the rules in the knowledge base are prepared, the kernel carries out meshing by a production system using the knowledge base. In addition, a data structure which is suitable for LISP is proposed in order to treat the complex data structure in the meshing processes. The triangular-mesh generator described in [20] consists of the kernel, knowledge-base, functions for the mesh generator, and data for the computational model.

The next three papers continue the theme of knowledge-based and data-based methods for CEM. Knowledge-based programs would appear to be able to provide a high-level capability, which, when coupled with conventional numerical analysis routines, can produce a computer system capable of aiding a designer in the exploration of design space. In fact, software tools have been developed [21] which permit an experienced designer to transfer his expertise regarding the analytical model of a device. The interesting factor is that the transfer and the execution of knowledge are both performed explicitly, that is, symbolically. Knowledge is declaratively represented instead of programmed procedurally. This facilitates the creation of design tools, usually referred to as "expert systems," to be used by non-expert designers [21].

One component of knowledge that lacks explicit representation in present-day expert systems for magnetics is that of bounds on parameter values [21]. In realistic designs, parameters rarely have definite values associated with them; rather, they are likely to be given along with some tolerance, or perhaps a discrete set of allowable values, as, for example, lamination thicknesses in an electrical machine. Although an expert designer may impose restrictions on the possible range of values a parameter may have, the complete implications of these restrictions are usually unforeseen by the expert, and this may lead to an inconsistent knowledge base. If a design is tested simply in a numerical analysis system, it is unlikely that any inconsistencies in the specification, such as an impossible lamination thickness, would be identified–it is the job of the expert designer to recognize this problem. Propagation of interval labels (interval inference) can enhance the design process by uncovering such inconsistencies at the modeling stage. Similarly, end-users would benefit from interval specifications on design parameters because the solution space for valid designs is effectively reduced. The value bounds provide the non-expert with information allowing him to proceed towards the desired goal much more quickly.

Paper [21] describes a calculus of interval mathematics and applies it to the development of an expert system for computer aided design of electromagnetic devices. Interval mathematics was introduced in 1966, and is now a well-established subject [21]. It has traditionally been used for bounding errors due to machine arithmetic and errors propagated from inexact data in numerical methods. The premise is that the computed interval with finite precision endpoints is a bounding box for the actual infinite precision result. A complete package that performs interval mathematics must be capable of performing set-function operations, such as union and intersection, as well as the usual mathematical (algebraic) operations. The formulas given below are typical of interval mathematics:

$$\begin{aligned} A+B &= [a_1+b_1, a_2+b_2] \\ A-B &= [a_1-b_2, a_2-b_1] \\ A\cdot B &= [\min\{a_1b_1, a_1b_2, a_2b_1, a_2b_2\},\ \max\{a_1b_1, a_1b_2, a_2b_1, a_2b_2\}] \\ A/B &= [a_1, a_2]\cdot[1/b_2, 1/b_1], \end{aligned}$$

where $A=[a_1,a_2]$, $B=[b_1,b_2]$, and $/$ is not defined if $0\in[b_1,b_2]$.

The use of interval techniques is not restricted to expert systems, however. They may also find applications in low-level numerical algorithms for electromagnetics [21].

When finite-element, or finite-difference, methods are used to solve open-boundary problems, one has the added inconvenience of establishing a "boundary at infinity" to close the solution space. This causess a lot of wasted computation, so if one can establish an efficient way of arranging the computations, one will have gained something. This is especially true if the scattering body (or the object under consideration) is changed; there need not be a corresponding change in the conditions at infinity. Hence, if the problem is solved once, one may be able to establish a data-base that represents the conditions at infinity for other variations of the problem. Paper [22] describes such an arrangement.

In [23] a unified data-base (DB) for data exchange among field computation and CAE programs is presented. The DB consists of four units, by function:

1. The **pre-processing** unit contains:
    - a commercial solid modeling system called Geomod. In Geomod, objects can be created by constructive solid geometry, by sweeping and skining techniques.
    - a commercial mesh generating system, Supertab, which is oriented towards mechanical engineering. Supertab is capable of meshing complex geometrical shapes and sculptured surfaces.
    - an automatic mesh generator for electromagnetic field design. This generator works on polyhedral geometries, and is capable of meshing non-convex regions.
    - an object manager which provides a set of operations for maniupulating data stored in the database. These operations enable information to be treated as *objects* instead of FORTRAN arrays. This frees users from the internal representation of objects, and enables designing at a high level of abstraction.
2. The **processing** unit contains three established field processors. They are:
    - Flux3d, which uses the finite element (FE) method. It solves non-linear and linear magnetostatic, electrostatic and magnetodynamics problems.
    - Phi3d, which uses the boundary-integral equation (BIE) method. Phi3d can solve linear electrostatic, magnetostatic and high-frequency magnetodynamics problems.
    - Trifou, which uses a mixed FE-BIE method. Trifou uses edge elements for solving linear, low-frequency magnetodynamics problems.
3. The **post-processing** unit is made up of two post-processors. Current efforts consist of developing a common post-processor for all three field codes.
4. The **data administration** unit is an interactive data-structuring module, through which the data administrator creates and maintains the basic TDS. (TDS is an abbreviation for a unified data structure called the TRIFLUX data structure, which is based on the characterization of operational data used in the three field codes. It contains geometrical data structure, finite-element data structure, inductors, and material properties.)

This type of system is useful in solving integrated problems, such as those involving electromagnetic, structural, and thermal variables. These are precisely the same problem one faces in designing microwave tubes.

In [25] a mesh refinement algorithm for arbitrary tetrahedral meshes has been developed. The algorithm is suitable for use in a variety of adaptive mesh refinement schemes, and has the following features: (1) it can be applied to both optimal (Delaunay) and non-optimal meshes, (2) new nodes are inserted using a "perturbed" edge bisection to prevent crossing edges, and (3) the Delaunay ceiterion is applied locally over each tetrahedron selected for refinement. the advantage of the local Delaunay subdivision is that it decouples the subdivision process which reduces computation time. The method has been successfully applied to several magnetostatic problems modeled using first-order tetrahedra, and has produced refined meshes of over 215,000 elements.

The mesh refinement process consists of three steps: First, the areas of the field region requiring refinement are identified either manually or automatically using a suitable error criterion. Second, the elements in the identified areas are subdivided in a manner which preserves the essential fetaures of the original mesh. Finally, the refined mesh is solved using the finite-element method. This process is repeated until the required accuracy is achieved or the computational limits are reached.

The first-order tetrahedra used in [25] maintain continuity of the normal component of magnetic induction between elements, but not the tangential component of the magnetic field intensity. The violation of the continuity of tangential H at the element interfaces is used as a measure of the local smoothness of the field, from which the error measure for remeshing is derived. Two forms expressing this smoothness are examined. The first is in the form of a current sheet at the element interfaces, which is a consequence of the discontinuity in tangential H, and the second is an energy measure obtained by "weighting" these current sheets with the corresponding vector potential solution.

In [26] the mesh generation process for two-dimensional problems is discussed, and a new mesh generator based on expert-system methods is presented. For every corner of a closed polygon, the corner-angle conditions are analyzed and different types of mesh primitives are generated. In every step of the mesh generation process, the conflict with the existing mesh is recognized, and an optimal action is chosen. Thus, any area surrounded by a polygon even with a very sophisticated contour can be successfully covered with a basis mesh. This mesh is further improved by side-swapping and node-shifting algorithms, both optimizing the shape of the elements and the location of the nodes. Problem-oriented adaptive mesh refinement algorithms provide further optimization according to suitable criteria.

The rules for forming the mesh are:

- The number of elements should be as small as possible for a desired accuracy of the solution.
- The triangles should differ as little as possible from equilaterals. This provides the required high quality of the approximation.
- A constant decay of the density of elements should be provided to bridge the gap between areas with a coarse density of node-chains and areas with high density.

Existing mesh generators may be classified into two categories: Interpolation mesh generators, which construct a mapping from a canonical domain onto the structure to be analyzed. These require an initial gross partitioning of the structure into simpler sub-blocks, and also have difficulty in carrying out local mesh refinement [27]. The second category, automatic triangulating mesh generators based on Dirichlet tessellation and dual Delaunay triangulation, work well for two-dimensional applications, and easily accommodate local mesh grading, but have problems, such as slivers, degeneracy, and exterior boundary alteration for three-dimensional applications [27]. These generators also rely on user-supplied nodal points on the surface of the structure, and have diffulty imposing local mesh restrictions which are not compatible with the Delaunay criteria [27].

Researchers have turned to artificial neural networks in an attempt to reduce these excessive demands. Work in the area of neural networks has demonstrated simple, self-organizing, topology-preserving mappings that have asymptotic characteristics quite similar to the Dirichlet tessellation. The authors of [27] present an automatic mesh generator, SOFT (Self-Organizing Finite-element Tessellation), that is based on self-organizing neural networks. Both interpolation mapping and Dirichlet tessellation characteristics are asymptotically incorporated in the new mesh generator.

A new technique for three-dimensional mesh refinement and adaptive mesh generation is presented in [28]. A procedure for refining three-dimensional tetrahedral meshes, based on Delauney criteria, is developed. Additional nodes are inserted on an existing mesh, and the tetrahedra produced are transformed, so that an optimum mesh is formed. Solution of a problem with an initial coarse mesh is followed by successive refinements. Furthermore an *a posteriori* error analysis is employed to estimate local errors and refine the mesh at those regions. The discontinuity of the normal field gradients on common interfaces, is proposed as a criterion for error estimation. Several different examples using the proposed technique are presented to illustrate the method.

There are two types of adaptive methods available in finite-element analysis, h-type and p-type. In h-type adaption, with which the previous papers were concerned, the degrees of freedom are added to the mesh by re-meshing, i.e., by building another mesh with smaller finite elements in some places. The name stems from the fact that the maximum dimension of a finite element is usually designated with an h. In p-type adaption, on the other hand, only one mesh is used [29]. The extra free parameters are added by raising the polynomial order of some of the elements. This is only possible if the elements are hierarchal in nature, i.e., if mixing of polynomial orders within the same mesh is permitted [29]. An advantage of p-type adaption is that it avoids re-meshing. In two-dimensions, fairly efficient algorithms are available for automatic mesh refinement; nevertheless, re-meshing can be expensive. In three-dimensions the difficulties and expense of mesh generation are much greater. For this reason alone, it is worth pursuing the development of p-type methods.

A further advantage of p-type adaption applies especially in the case of high-frequency problems. Away from singularities, increasing polynomial order (p-refinement) gives a better rate of convergence than using a larger number of fixed-order elements. In low-frequency problems, a high density of degrees of freedom is needed near singularities, but lower densities are needed further away, in homogeneous regions, where the field becomes increasingly uniform. Thus, the discretization is determined to a large extent by singularities–sharp corners, thin sheets of current or charge, and so on. Consequently, p-refinement has less to offer low-frequency analysis. On the other hand, homogeneous regions in microwave and optical devices still have considerable, wavelike, field variations, and require a relatively high density of degrees of freedom. It is to be expected that p-type adaption would be particularly effective for such devices. Paper [29] explores p-type adaptive schemes for high-frequency analysis, using hierarchal, curvilinear, triangular elements.

As we have seen, the generation of three-dimensional meshes for finite element analysis can be accomplished in various ways. Since the application should not be restricted to a few geometries, a method is proposed in [30] which is based on a surface model of the problem's geometry. The geometry is uniquely defined by the set of the subregion surfaces. Many electromagnetic problems, especially those concerned with electrical machines, can be described using a few surface shapes. Therefore the program, which does the surface triangulation, is small and can easily be expanded, if a special shape is needed [30].

The three-dimensional decomposition proposed in [30] is accomplished by a mesh generator which triangulates three-dimensional subregions into tetrahedra, using only a given surface triangulation produced in a prior step. The decomposition scheme does not need to be changed if the class of usable geometries is expanded by additional surface shapes, yielding easy program maintenance and high reliability. The method can be applied to every three-dimensional problem, while

methods using two-dimensional triangulation with expansion to layers in the third dimension or with three-dimensional meshes generated using a composition of basic volumes (cubes, spheres, etc.) are restricted in their application.

The mesh generated by the method proposed in [30] is coarse, and some of the tetrahedral elements can be of bad shape, but these properties can be accepted, because the decomposition is intended to be a starting mesh for an *adaptive* mesh refinement process [30].

Self-adaptive methods free the user from the requirement of attaining a significant level of expertise in the application of the finite-element technique. They allow solutions of problems to be automatically generated to a pre-specified level of tolerance. The key to the method is the ability to estimate the distribution of the error associated with a particular solution, and then to use this distribution to target those areas of the problem so as to reduce the local error [32]. Complementary functionals have been used in a variational principle to provide error bounded solutions to two-dimensional and axisymmetric three-dimensional static field, and eddy-current problems, and to provide a basis for a fully self-adaptive meshing procdeure for such cases [32]. The extension to three-dimensions is not trivial, there being a great increase in complexity associated with the generation of the appropriate error-bounding functionals and the applicable boundary conditions. Also, the geometric problems associated with the management of a three-dimensional mesh refinement procedure for multi-material problems are many. Paper [32] addresses these matters.

The idea behind the complementary functionals is to define a "primal standard form" variational problem with the vector potential as the unknown, and its associated boundary conditions. The "primal complementary form" uses two scalar potentials, which generate the magnetic field intensity, as the unknowns, together with their boundary conditions [32]. Expressing the difference between the complementary field solutions, over each element, as a percentage of their mean provides an estimate of local error which can be used as a basis for mesh refinement. To begin with, however, it is necessary to generate the initial mesh. For general expediency, and with the needs of automatic refinement in mind, this is done in [32] using tetrahedral elements.

The three options considered in [32] for the mesh generation phase were, Delauney Tessellation, the finite octree technique, and the emerging approach of specifying a desired element size throughout the mesh (see [13] for an example of this approach). A Delauney Tessellation was considered to have a number of advantages, and adopted as the basis for mesh generation. The principal advantage is a capacity for automation, and this is especially important in the context of a self-adaptive environment [32]. A full, valid, finite-element mesh can be constructed from a Delauney Tessilation, working solely from a geometric model [32].

In [33], the authors merge their novel mesh control idea with a constrained Delauney triangulation algorithm, and an algorithm for the initial triangulation, to develop an efficient, flexible, and reliable triangulation of complicated solids [33]. A new strategy for adaptive mesh refinement is also included.

The Delauney triangulation is desirable for finite-element analysis because of its optimal properties. The approach to automatic mesh generation using Delauney triangulation proceeds by first generating points within and on the boundary of a solid model, followed by forming a triangulation considering the points one-by-one. Even though a number of algorithms have been published, the following problems have not been adequately solved [33]:

1. **Constrained Delauney Triangulation.** In finite-element applications, the problems are usually multi-material ones, which means that triangulations must be constrained by the boundary and interfaces of the problem. We call the Delauney triangulation an optimal constrained triangulation. Those two-dimensional algorithms that are based on it do not work in three-dimensional cases, and, in fact, some 2D algorithms have troubles with constraints.

2. **Degenerate Problem.** Careful attention must be paid to the so-called degenerate and near degenerate problems, even in those algorithms that are currently used.

3. **Efficiency.** Since a lot of search and calculations are needed to determine that element to which a new point is to be added, as well as the location of that point, the amount of time required in a Delauney-based algorithm can be large. In fact, the asymptotic time growth rate is greater than linear.

4. **Node Spacing.** The usual triangulators consist of two independent procedures: node spacing, and triangulation. The triangulation process is automatic, so that the degree of automation and mesh quality fully depends on the node placmement method, i.e., on controlling the mesh. One of the most important open issues is how to space nodes as desired most efficiently, particularly for three-dimensional cases.

5. **Smooth Refinement.** The usual methods cannot refine a mesh smoothly. For example, a mesh of NE elements cannot be adequately subdivided into, say, $2.4NE$ elements. Therefore, during a self-adaptive finite-element analysis, an existing mesh cannot be well refined, as may be required by the error estimation algorithm, and more adaptive steps will be needed.

The mesh control idea proposed in [33] is another one based upon the size of the elements, which the user is allowed to edit interactively. The scheme differs from other Delauney-based methods, in allowing the node-spacing and triangulation steps to process simultaneously and adaptively. Because of this, the mesh generation asymptotic time growth rate is linear [33].

One of the main obstacles to the general use of finite elements is the difficulty of obtaining a mesh of suitable quality for a magnetic device, such that an "accurate" solution is obtained [34]. Currently, most finite element analysis packages require that the level of discretization be specified manually. This means that the user of the package must have a certain level of expertise in finite elements and in magnetic field analysis. One method of overcoming this limitation involves the use of some type of adaptive meshing scheme to guarantee sufficient accuracy in the finite element solution. While this approach is attractive, it can be slow to converge since it starts with zero knowledge of the solution. In effect, the adaptive system "learns" about the solution as it proceeds [34].

Using a few rules a human expert can determine a good distribution of elements from a description of the magnetic device alone. For example, sharp corners in iron require a relatively high mesh density in order to accurately model the solution near the singularity. It appears that the expert is capable of applying "prior," or already learned, knowledgeto the meshing process; the result is an improvement in the convergence of the adaptive solution to a "good" mesh. However, the rules used by the expert are difficult to formulate in the frqmework of conventional expert systems, because they involve metric features and material properties, concepts which are not easily included in "if-then" type rules. Paper [34] introduces a new approach to this problem using neural networks.

The neural network concept is a computing model that is characterized by the ability to generalize from examples, to extract features present in a set of inputs, to degrade smoothly, and to tolerate uncertainty [34]. The back-propagation neural network was selected for the application described in [34]. This type of network computes a mapping from its inputs to its outputs. This mapping is learned from examples of the required mapping, thus avoiding the difficult process of knowledge acquisition from human experts [34].

The steps needed in the application of neural networks to the meshing problem are [34]:

1. **Representing the Magnetic Device.** The input to the neural network must represent the magnetic device, and the output the element size distribution. The input and output must be in the form of fixed-length vectors of bounded real values. The main property of the meshing problem is that the mesh at a given location is largely dependent on only the *local* device features.

2. **Representing the Element Size.** The output must represent the element size required at a specified point. Good performance is achieved using proportional coarse coding, in which a number of outputs are used, with each output representing a discrete element size. The example sizes are encoded by assigning to each output the probability that the element size would fall in its range, using a Gaussian probability distribution centered at the size and with a standard deviation of half the range of each output.

3. **Generating the Training Examples.** The neural network learns the input-output mapping from examples. In this case, the examples are taken from "ideal" meshes of representative magnetic devices. (An ideal mesh is one in which all triangles have the same error in the vector potential.) Since the neural network computes the element size at a point, a single device can be sampled at may points to give many training examples. If the device is designed to include the main features of magnetic devices in general, than all examples can be taken from one device.

4. **Training the Neural Network.** This application [34] uses a four layer, fully connected neural network topology with 8 nodes in the input layer, 24 nodes in the first hidden layer, 18 nodes in the second hidden layer, and 10 output nodes. This topology contains a total of 856 weights, and gives the best performance among various other networks with between one and three hidden layers, and with both larger and smaller hidden layer sizes. Also, because very different rules are used to mesh air and iron, two separate networks are used, one trained to compute the element size at a point in air, the other at a point in iron. Both networks have the same topology, but different weights.

The increase in computational power of current programs allows the user to handle problems of growing complexity. However, this improvement in modeling capabilities leads to increasing difficulties when specifying problem characteristics and it becomes necessary to develop user-friendly CAD tools along with computer programs. The authors of [35] present three software tools that were developed, together with an electromagnetic field analysis program, within the scope of finite-elements and boundary-elements (and a combination of the two). The first tool consists of a two-dimensional or three-dimensional structure generator whose results are transmitted to a second tool, which is an interactive meshing program or an automatic surface generator.

The graphical language interpretor that was developed in [35] offers the possiblility of defining local and global variables and arrays, to call recursive subroutines, to use 'do' and 'if then else' loops, to introduce comments, to access classical arithmetic functions. The programmer can also create elements, operate combined translations, rotations and symmetries on parts of the structure, and to work in major coordinate systems. Those instructions were specialized to the geometry at hand, i.e., whether two-dimensional or three-dimensional problems were being solved. There are some difficult tasks faced in three-dimensional meshing; it is sometimes more difficult to generate surfaces that will support finite elements, than to create the mesh in the first place. This is particularly true when finite-elements are coupled to boundary-elements, because an internal mesh of nonlinear materials, only, is required.

The tool described in [36] utilizes the object oriented language, SMALLTALK, to allow descriptions of device classes, parameterizations and graphical-aided creations and modifications, use of

part-whole hierarchy and multiple inheritance, in a highly uniform environment. It also serves as an interface to finite-element packages.

Although field computation in two-dimensional axially symmetric problems has a long-standing history, and commercial software for such problems has been available for many years, and despict the big increase in the computer power in the last decade, its practical application for iterative design is limited by the amount of user interaction required. Most designers would welcome a practical design tool for their devices (very often limited to one device) in place of the general purpose finite element packages currently available. Most of them want programs which are specific to their needs. An outer shell can give designers all they want and hide all complexities of the general purpose FEM package, leaving all of its power in hand for when it is really needed [37].

In [37] a special shell for a general purpose FEM package to assist in the CAD of axisymmetric devices, such as loudspeakers, is presented. The basic idea of the shell is that a special script file may be quickly and easily created for every design. In this file everything needed to solve a particular problem is stored, for example, the geometry of the problem, basic information about the mesh, problems definition, etc. Then the whole solution procedure is conducted automatically, i.e., the mesh process, problems definition, solving and post-processing. Then if the designer wants to change anything in the design (say one dimension) only one piece of data has to be changed using an interactive graphics editor to run the whole FEM package again. Moreover, the designer can prepare a sequence of such tasks and run them automatically without any user intervention. For any specific area of interest some general geometries can be built into the shell. For example, a loudspeaker designer can simply choose one of the standard loudspeaker geometries, and there is still the possibility of drawing a completely new one [37].

Practical three-dimensional electromagnetic modeling has traditionally been hampered by insufficient computing power. Recent advances in computer hardware are beginning to remove this difficulty, and one can expect that this trend will continue for the foreseeable future. To exploit this increased computer power and to solve more realistic problems, researchers have developed numerous algorithms appropriate for 3D calculations, and have built large general-purpose computer codes around them. In spite of this technical and theoretical progress, two practical difficulties remain: providing the computer code with an accurate description of a particular problem (pre-processing) and viewing the end results of the calculation (post-processing). These become duanting tasks for all but the simplest of problems in 3D due to the large amount of data involved [38].

Researchers at the Lawrence Livermore national Laboratory have been developing and using a new finite-difference, time-domain (FDTD) code over the last few years. This code TSAR, is currently being used on a wide range of electromagnetic scattering, coupling, and propagation problems. Some of the geometries of interest are large and quite detailed, requiring meshes with more than a million cells. In addition, an FDTD code is often run for thousands of time steps, producing an enormous quantity of output data [38].

To efficiently deal with these large problems, the authors of [38] have developed a set of pre- and post-processing tools to be used in conjunction with the TSAR FDTD code. This set of utilities consists of a solid-model based mesh generator, a mesh verifier, and a color/surface plotter. These tools all run on graphics workstations, and, due to their highly interactive nature, are quite easy to use. For added convenience, some of the workstations are connected to a videotape system. Whith this arrangement, users can record complex time-varying results in a convenient and portable format [38].

The automatic mesh generator being developed at LLNL is based on a solid modeling approach, in which the user builds a mathematically precise representation of the undiscretized object using a solid modeling program. A special-purpose mesh generator code then operates on the solid-model database to automatically produce an FDTD grid. The code attempts to take into account the

spatially-staggered nature of the FDTD mesh and the vector nature of the unknowns. This yields a faster and more reliable process than before, and users need build their models once, independent of cell size. If a finer mesh is required, only the mesh generator needs to be rerun. This provides a means of true mesh *generation* as opposed to mesh *construction* [38].

An interactive visualization code, that runs on a high-end color workstation, allows the user to rotate, slice through, and zoom in on various portions of the mesh, in near real-time. In addition, various materials and/or field components can be displayed in different colors or erased to aid in the inspection of a complex mesh [38].

A variety of post-processing utilities, that are highly interactive and based on workstations, have been developed for TSAR. With these facilities, the user can display a 2D slice of data, either as a surface plot or as a color fringe plot. Two dependent variables can be displayed simultaneously using both color and height. The pre- and post-processors use a mouse/button user interface [38].

In finite element analysis, discretization error plays a major role in determining the accuracy of the final solution. Solution accuracies in the range of 5% to 15% are acceptable for most engineering applications, but for some applications very high accuracy in the range of 2% to 3% is required. In order to improve the accuracy of the solution adaptively, an error estimate should be defined, that is reliable enough to identify the critical regions of the domain that have the larger errors. This requires an efficient and quantitative error estimate to accurately gauge the error in the solution. Thus, the efficiency of an adaptive finite emelment computation depends upon the availability of a computationally robust and reliably stable error estimate. The reliability of an error estimate provides a measure of accuracy of the computed solution. Reliability analysis for electromagnetic field problems using two different *a posteriori*[3] error estimates is proposed in [39]. A mathematical model is used to define reliability analysis in the first part of the paper, while in the second part the theory of reliability assessment of an error estimate through asymptotic exactness is outlined. Numerical test results for a sequence of nearly optimal adaptive meshes[4] for a 2D problem are presented in the third part.

Frequently, one must terminate a mesh that is to be used in the finite element solution of an electromagnetics problem, with the result that a boundary-integral equation is introduced. The derivation of *a posteriori* error estimates for a coupled finite-element boundary-element problem is the subject of [40]. A brief summary of the variational boundary-value problem formulation of the 2D finite element–boundary element (FE/BE) method is presented. From this *a posteriori* error estimates and error indicators for the FE/BE method are developed and applied to electromagnetic scattering and radiation problems. The results obtained indicate that these error estimates and indicators can be obtained within negligible computational times and can be successfully to obtain valuable *a posteriori* accuracy and convergence information regarding the reliability of the FE/BE method solutions.

A reliable *a posteriori* error estimate for a FE/BE method solution would enable one to obtain valuable convergence information without having to solve the same problem using a *larger* number of unknowns. The error estimate can thus be used as a convergence check for practical electromagnetic problems for which no analytical solution exists. This would be especially advantageous when

[3]The expressions '*a priori*' and '*a posteriori*' often appear in error analysis. An *a priori* error estimate is one that is theoretically known before a computation is made. The coefficients in the expansion for the error estimate, however, may not be known. Such error estimates may not be precise enough to be used in practice without making some computations, i.e., performing a numerical experiment. The resulting error estimate then becomes *a posteriori*, that is, after the experiment is performed.

[4]A mesh is said to be optimal when the measure of error in the solution is equal for each element in the mesh. Those measures of error that are commonly used are the energy norm, relative percentage energy norm error, and local and global effectivity indices. The effectivity index, $\Theta = \|e\|/\|e\|_{ex}$, where $\|e\|$ and $\|e\|_{ex}$ are computed and exact errors, respectively, in the energy norm.

electromagnetically large problems, that reach the practical solution time and memory limits of the computer at hand, are considered. This is true regardless of the computational power available [40].

Meyer and Davidson [40] use the Element Residual error estimate Method (ERM), which provides *a posteriori* error estimates after two solutions have been obtained. The first solution, E^1, utilizes lower-order polynomial basis functions, whereas E^2 is based on higher-order polynomials[5]. In applying the ERM, one first forms the relative electric field error

$$E^{12} = E^2 - E^1 \ ,$$

from which the global electromagnetic energy norm (EM-norm) is formed:

$$\left(\|E^{12}\|_{\Omega}^{EM}\right)^2 = \sum_{k=1}^{M} \left(\|E^{12}\|_{\Omega_k}^{EM}\right)^2 \ .$$

Ω is the domain of the problem, Ω_k is the region occupied by the kth finite element, M is the number of finite elements, and

$$\left(\|E^{12}\|_{\Omega_k}^{EM}\right)^2 = \int_{\Omega_k} \left(\mathrm{Re}\left(\frac{1}{\mu_r}\right) |\nabla E^{12}|^2 + \mathrm{Re}\left(\epsilon_r k_0\right) |E^{12}|^2 \right) d\Omega_k$$

is the EM-norm of the actual error associated with finite element Ω_k. An estimate of this error can be obtained by solving a finite element problem on Ω_k, from which one can obtain an estimate of the global error [40]:

$$\int_{\Omega_k} \left(\mathrm{Re}\left(\frac{1}{\mu_r^k}\right) \nabla \Phi_k^{12} \cdot \nabla v^{12} + \mathrm{Re}\left(\epsilon_r^k k_0\right) \Phi_k^{12} v^{12} \right) d\Omega_k =$$

$$\int_{\Omega_k} r_k^1 v^{12} d\Omega_k + \frac{1}{2} \sum_{l=1}^{3} \int_{\Upsilon_{k(l)}} \frac{\partial E_k^1}{\partial n_{k(l)}} v^{12} d\Upsilon_{k(l)} +$$

$$\int_{\Upsilon_k \cap \Gamma^{N_1}} \left(g_1 - \frac{\partial E_k^1}{\partial n_k} \right) v^{12} d\Upsilon_k \quad \text{for all } v_k^{12} \ .$$

In this equation v_k^{12} is the weighting functions on Ω present in the FEM solution E^2, but not in the solution E^1. $\Upsilon_{k(l)}$ is the side of the triangular finite element Ω_k connecting Ω_k with its lth neighboring element $\Omega_{k(l)}$, r_k^1 is the local element residual of the governing equation associated with E^1, and $\partial E_k^1 / \partial n_{k(l)}$ is the jump in normal derivative between element Ω_k and its neighboring element, $\Omega_{k(l)}$.

Some typical results that indicate the convergence of the solution, as well as computational times are included in Tables 7 to 9. It is important to understand that E^2 needs to be computed only if one wishes to calculate the *actual* error $\|E^{12}\|$. This is an expensive calculation, as Tables 8 and 9 indicate. The computation of the estimate, $\|\Phi^{12}\|$, however, is quite inexpensive, as indicated in the fourth column of Tables 8 and 9.

These authors conclude that adaptive finite element methods are closely linked to *a posteriori* error estimates, and could be used to improve the efficiency of general FEM solutions. The *a posteriori* error estimate methods can be used to identify the regions where the fields need to be approximated more accurately (for example where the fields vary more rapidly), and the finite

[5] Clearly, this is an example of an *hp*-formulation; h is the maximum mesh size, and p is the order of the polynomial basis.

$M : M_b$	Estimated $\frac{\|\Phi^{12}\|^{EM}}{\|E^1\|^{EM}}$	Actual $\frac{\|E^{12}\|^{EM}}{\|E^1\|^{EM}}$
E^1:first-order basis;E^2:second-order basis		
170:24	79.99	> 100
501:43	74.38	79.30
977:73	51.42	51.17
1502:90	44.09	35.61
E^1:second-order basis; E^2:third-order basis		
170:24	47.99	74.48
501:43	11.34	13.38
977:73	7.04	7.35
1502:90	5.52	4.24

Table 7: Percentage error estimates compared to actual errors. M is the number of finite elements, and M_b is the number of boundary elements. Taken from [40].

M	Computational time (hours:min:sec)		
	FE/BE method solution		Error estimates
	E^1	E^2	
170	00:00:02	00:00:10	00:00:05
501	00:00:06	00:01:33	00:00:10
977	00:00:23	00:08:59	00:00:21
1502	00:00:42	00:18:28	00:00:29

Table 8: Computational times on an HP-720 workstation. E^1 are first-order basis functions, and E^2 are second-order. Taken from [40].

M	Computational time (hours:min:sec)		
	FE/BE method solution		Error estimates
	E^1	E^2	
170	00:00:10	00:01:25	00:00:07
501	00:01:33	00:12:27	00:00:14
977	00:08:59	01:28:58	00:00:29
1502	00:18:28	02:27:31	00:00:40

Table 9: Computational times on an HP-720 workstation. E^1 are second-order basis functions, and E^2 are third-order. Taken from [40].

element mesh can thus be adapted to ensure superior basis function distributions in these regions. The 2D methods described in [40] can be extended to 3D formulations that use edge-based elements, though hierachical edge-based elements may prove more formidable than the nodal-based equivalents if the error estimator is used to drive an adaptive meshing algorithm.

An *hp* adaptive analysis is described in [41], in which a Delaunay-based automatic quadrilateral mesh generator is used. Together with the Zienkiewicz-Zhu *a posteriori* error estimator [42], this mesh generator meets a prescribed accuracy within one, or at most two, *h*-refinement stages with optimal convergence rate. Furthermore, one achieves a simple and rapid improvement in the solution by uniformly increasing the element interpolation order, *p*. In [41], the *h*-refined mesh was taken as the base for *p*=refinement, and the *p*-refinement is uniform throughout the entire mesh.

The advantage of the *hp*-refinement strategy is based upon the observation that the asymptotic convergence of the finite element approximation is given by

$$\|e_h\| = O(h^{\min(p,\lambda)}) \ ,$$

where p is the order of the polynomials used in the shape functions, λ is a constant (< 1) typical of the singularity in the solution, if there is one, and h is a characteristic size of the element. From this observation, one can show [41] that

$$1 - \left(\frac{1}{2}\right)^{\min(p,\lambda)} \leq \Theta \leq 1 + \left(\frac{1}{2}\right)^{\min(p,\lambda)} \ .$$

We can conclude, therefore, that in uniform *h*-refinement, the effectivity index of the error estimator deteriorates when the solution is dominated by a singularity (as, for example, in the third canonical problem of Chapter 3). Under this condition, the effectivity index is independent of the order of the polynomial, p [41]. When the sequence of *h*-refined meshes is optimal, however, i.e., when the error is approximately equally distributed over each element of the mesh, then it can be shown that [41]

$$1 - \left(\frac{1}{2}\right)^{p} \leq \Theta \leq 1 + \left(\frac{1}{2}\right)^{p} \ ,$$

and the dependence upon the singularity is avoided. Under this condition the effectivity index will tend to unity as $p \rightarrow \infty$.

In uniform *p*-refinement, the convergence is given by

$$\|e\| = O(N^{-\beta}) \ ,$$

where N is the number of degrees of freedom, and β depends upon the smoothness of the solution. The convergence reate is at least twice that of the uniform *h*-refinement, when the solution is smooth. When the mesh refinement is a proper combination of both *h*- and *p*-type, then an exponential convergence rate is possible, and we obtain an estimate of the form

$$\|e\| = O(\exp(-\gamma N^{\theta})) \ ,$$

where $\theta \geq 1/3$. Clearly, the *hp*-version of the finite element method is most effective in terms of the number of solution variables for general elliptic boundary value problems.

hp-refinement procedures are attractive, because *h*-refinement alone will generally require an excessive number of degrees of freedom to achieve high accuracy. The implementation of a local *p*-refinement scheme, however, is often computationally difficult. In [41] a uniform *p*-refinement is applied to the optimal *h*-refined mesh. For *h*-refinement, the Zinekiewicz-Zhu [42] error estimation method is used.

Usually, the user's accuracy requirement is stated in terms of an overall percentage error in energy norm, i.e.,

$$\eta = \|\mathbf{e}_h\| / \|\mathbf{A}\| \leq \eta_{ep} \ .$$

In order to achieve the optimal solution, each element should have the same permissible error, $\|\mathbf{e}\|_{ep}$. Let $\|\mathbf{e}_h\|_i$ be the energy norm of the estimated error associated with the ith element ($1 = 1, \ldots, NE$), and note that

$$\|\mathbf{e}_h\|^2 = \sum_{i=1}^{NE} \|\mathbf{e}_h\|_i^2 \ ;$$

then the maximum permissible element error is

$$\|\mathbf{e}\|_{ep} = \eta_{ep} \left[\frac{\|\mathbf{A}_h\|^2 + \|\mathbf{e}_h\|^2}{NE} \right]^{1/2} \ ,$$

which can be calculated after each solution. With this information, we can predict how to refine an element. Suppose

$$\xi_i = \|\mathbf{e}_h\|_i / \|\mathbf{e}\|_{ep} \ ;$$

then the predicted size of element i should be

$$h^i_{\text{new}} = h^i_{\text{old}} \ \xi_i^{\max\left(\frac{1}{\lambda}, \frac{1}{p}\right)} \ .$$

The preceding paper dealt with two-dimensional problems. Golias and Tsiboukis [43] point out that only recently have several significant contributions appeared in the area of adaptive refinement in three-dimensional problems. The delay has been probably due to the difficulty in developing geometrical algorithms for refining 3D meshes, but with the proposal of several techniques [44] [45] [25], indications are that the problem is being successfully attacked.

Edge element are used in [43], which means that the unknowns are the circulations of the vector field around the edges of the tetrahedron. In this way, unknowns are related to the edges of the mesh, and edge elements impose tangentical continuity of the electric field, $\mathbf{E}$, but not normal continuity. Thus, edge elements do not impose over-continuity, as is the case with vector nodal elements.

Two new error estimators for eddy current problems are presented in [43]: (a) the tangential discontinuity in the magnetic field, $\mathbf{H}$, and (b) the normal discontinuity in the eddy current density $\mathbf{J}_e$. Consider two tetrahedra (e) and (f), and their common face (ABC). Let $\mathbf{H}^{(e)}$ and $\mathbf{H}^{(f)}$ denote the magnetic field in tegrahedra (e) and (f), respectively. The error estimator of the magnetic field $\mathbf{H}$ of face (ABC) is defined by:

$$EI^H_{(ABC)} = \int_{S_{(ABC)}} |\mathbf{n}_0 \times \left(\mathbf{H}^{(e)} - \mathbf{H}^{(f)} \right)|^2 dS \ .$$

Let $\mathbf{J}_e^{(e)}$ and $\mathbf{J}_e^{(f)}$ denote the eddy current density in tetrahedra (e) and (f), respectively. The error estimator of the eddy current density $\mathbf{J}_e$ of face (ABC) is defined by:

$$EI^J_{(ABC)} = \int_{S_{(ABC)}} |\left(\mathbf{J}_e^{(e)} - \mathbf{J}_e^{(f)} \right) \cdot \mathbf{n}_0|^2 dS \ .$$

The flow chart of the adaptive refinement algorithm is shown in Figure 24. An initial mesh is generated, and boundary conditions are set on this mesh. Then the eddy current problem is solved,

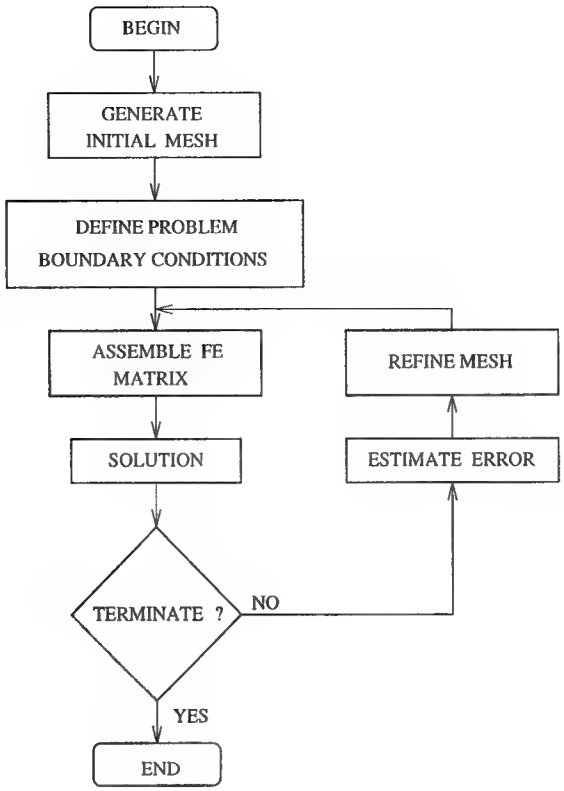


Figure 24: Algorithm for the self-adaptive procedure in [43].

followed by the calculation of the error by means of the above error estimator. The mesh is then refined in those regions that have high error. A threshold value is set, and all elements with error greater than this threshold are refined. The problem is re-solved, and this cycle of solution and refinement continues.

The error estimator of the magnetic field **H** is applied throughout the whole domain, whereas that for the eddy currents is applied within conducting bodies, only. The technique that was actually implemented in [43] for the combined application of the two error estimators is as follows: First, the magnetic field error estimator is applied, and refinement takes place in the whole domain. Then, extra refinement within the conducting regions is accomplished with the application of the eddy current density error estimator. In this way an even higher density of tetrahedra is obtained in conducting regions, thereby enabling a more accurate calculation of the distribution of eddy currents, and the approximation of the skin effect.

Our interest in this approach [43] lies in the fact that the authors are simultaneously solving for two different fields, **H** and $\mathbf{J}_e$, simultaneously, treating them as having distinct needs on the same mesh. This may be an approach that one could take in solving coupled TM and CEM problems on a single mesh, if the analysis codes for each field can accomodate a single mesh. Otherwise, one will have to interpolate between two different meshes, and this could complicate the simultaneous adaptation of two grids.

4.2 Extrapolation of Solutions

The error-estimating schemes described above, though useful when used in conjunction with adaptive mesh-generators, still do not give us an estimate of the final answer. They only inform us as

to whether we are converging, or not, and give us an estimate of the rate of convergence. If we have *a priori* information about the rate of convergence of a solution, we can use an extrapolation technique (Richardson extrapolation) to reduce the truncation errors due to the mesh-fineness, and give us an estimate of the 'exact' solution that would be achieved if the mesh-fineness were zero. We will demonstrate this procedure by referring to some one-dimensional examples of TOPAZ2D problems that are defined on a regular mesh. Though these examples are taken from [46], we point out that we have used the same extrapolation process to get improved accuracies in the calculation of impedances in eddy-current nondestructive evaluation with our proprietary volume-integral code, VIC-3D. Both Drayton's examples and VIC-3D use regular grids; this allows a straightforward application of Richardson extrapolation.

In Drayton's example, a slab of thickness L is excited by a temperature-dependent internal energy source $q_{\text{gen}} = K(A + BT)$, where K is the coefficient of thermal conductivity, T is the temperature, and A and B are arbitrary constants. The surface $x = L$ is maintained at temperature $T = 0$, and the surface at $x = 0$ is adiabatic. The solution to this problem is

$$T(x) = \frac{A}{B}\left[\frac{\cos(xB^{1/2})}{\cos(LB^{1/2})} - 1\right] .$$

Now assume that the truncation error in the finite element solution is $O(h^2)$[6], and that $A(h)$ is the computed solution at x_0. Then, if $f(x_0)$ is the true solution at x_0, we have

$$f(x_0) = A(h) + C_1h^2 + O(h^4) ,$$

where C_1 is indedpendent of the mesh size, h[7]. Now let $h = 2h$, and get

$$f(x_0) = A(2h) + 4C_1h^2 + O(h^4) .$$

Upon subtracting these two equations, and then rearranging, we find that, for sufficiently small h, the dominant error term is

$$C_1h^2 = \frac{A(h) - A(2h)}{3} ,$$

from which we conclude that an expression valid to $O(h^4)$ is given by

$$f(x_0) = \frac{4A(h) - A(2h)}{3} + O(h^4) .$$

Now, using TOPAZ2D we get approximations of accuracy $O(h^2)$ based on the step size $h_1 = 0.1$:

X Coordinate Position	Temperature Distributions Analytic	TOPAZ2D	Percent Error
0.000000	0.850816	0.846050	-0.560167
0.200000	0.813923	0.809373	-0.559018
0.400000	0.704714	0.700801	-0.555259
0.600000	0.527544	0.524647	-0.549153
0.800000	0.289476	0.287911	-0.540635

$O(h^2)$ approximations based on a step size of $h_2 = 0.05$ yield

[6]This can be theoretically justified.

[7]This is an example of an *a priori* theoretical error estimate. In order to determine the constant, C_1, we must compute the solution on two different grids. Then we get an *a posteriori* error estimate.

X Coordinate Position	Temperature Distributions Analytic	TOPAZ2D	Percent Error
0.000000	0.850816	0.849618	-0.140805
0.200000	0.813923	0.812779	-0.140553
0.400000	0.704714	0.703731	-0.139489
0.600000	0.527544	0.526816	-0.138000
0.800000	0.289476	0.289083	-0.135764

Note that the error when $h_2 = h_1/2$ is the mesh size is one-fourth the error with when h_1 is the mesh size, which justifies our assertion that the finite-element approximations are accurate to $O(h^2)$.

A single application of Richardson's Extrapolation Formula

$$D_2(h_1) = \frac{4A(h_2) - A(h_1)}{3}$$

yields an approximation of accuracy $O(h^4)$ as shown in the next table under the heading 'TOPAZ2D:'

X Coordinate Position	Temperature Distributions Analytic	TOPAZ2D	Percent Error
0.000000	0.850816	0.850807	-0.105781E-02
0.200000	0.813923	0.813914	-0.110576E-02
0.400000	0.704714	0.704708	-0.085141E-02
0.600000	0.527544	0.527539	-0.094779E-02
0.800000	0.289476	0.289474	-0.069090E-02

Note the significant improvement in applying Richardson's extrapolation: The error is reduced by more than two orders-of-magnitude from the solution with $h_2 = 0.05$.

Richard's extrapolation can, in principle, be applied to three-dimensional problems with complex meshes, but the computation of the solution when the number of degrees of freedom doubles can be a significant burden. That is the justification for using the *a posteriori* error estimators and adaptive meshing algorithms to improve the solution. With these techniques one selectively refines only parts of the mesh, rather than the entire mesh, in order to improve the solution. Still, as sophisticated as these numerical algorithms are, they do not allow one to extrapolate to get a solution that has an improved *a priori* error estimate.

5 IMPLICATIONS, RECOMMENDATIONS, AND CONCLUSIONS

'Grid-reusability,' that is the use or reuse of grid-related data depends upon problem conditions that are as diverse as geometry, the physical model, and materials properties. Nonlinearities that arise when a physical parameter depends upon the value of the field variable being sought can thoroughly complicate gridding requirements to such an extent that *a priori* guidelines are not available. The grid and its associated data will vary as the solution is computed iteratively.

The question of grid-reusability has surfaced only rather recently, as designers begin to address the question of concurrent engineering in a rigorous, self-consistent manner. We have shown a few examples in which researchers are addressing this aspect of CAE software development, but we are unaware of this problem being dealt with in commercial software.

The results of this study suggest that the conditions under which a grid and its related data can be used or reused are mandated by the physical model, as well as the geometry. As Canonical Problems No. 2 and 3 of Chapter 3 showed, the gridding requirments for a TM problem can differ considerably from those for a CEM problem, even if the geometry for each is a simple rectangle. The distinction between the two problems and their meshes lies in the nature of the boundary conditions and the nonlinear behavior of the material parameters; i.e., it is the physical model that distinguishes the needs of each problem. Indeed, the problem may be sufficiently complicated, that it is impossible to specify *a priori* guidelines for establishing a grid. The grid and its related data will vary as the solution is computed iteratively.

The major implementation problem in handling coupled analyses will be in supervising the transfer of data between the two meshes, and in adapting each mesh so that it becomes optimal for its respective problem domain. Little, apparently, has been done in the area of adaptive gridding for coupled problems, and we propose that Canonical Problem No. 3 could be the basis for a future research effort in this area. Even though the geometry for this problem is a simple rectangle, the physical models would present several areas where a grid would have to adapt in order to become optimal. The nonlinearity in the electrical conductivity might drive the grid adaptation, as well as the electrical singularity at the edge of the upper electrode. The thermal grid might have to adapt in response to the value of the convection coefficient. Canonical Problem No. 3 could also be the basis for further studies in *hp*-adaptive analysis of coupled problems, along the lines of [41] for EM problems.

The grand challenge of the immediate future is to extend the research reported herein to the problem of doing adaptive meshing on two or more grids for the purpose of optimally solving coupled problems with nonlinearities. What role will *hp*-adaptive analysis play in meeting this challenge?

References

[1] B. T. Chia, J. L. Coulomb, J. C. Sabonnadière, J. C. Vérité, and A. Nicolas, "A Unified Database for Data Exchange in Electromagnetic Field Computation," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 811-814.

[2] J. F. Ballay, F. Nuwendam, and J. C. Kieny, "A Tool Using an Object Oriented Language for Field Computation in a CAD Prospect," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1774-1777.

[3] G. Reyne, H. Magnin, G. Berliat, and C. Clerc, "A Supervisor for the Successive 3D Computations of Magnetic, Mechanical and Acoustic Quantities in Power Oil Inductors and Transformers," IEEE Trans. Magnetics, Vol. 30, No. 5, September 1994, pp. 3292-3295.

[4] H. Bohn, B. Giesen, A. Belov, N. Berkhov, E. Bondarchuk, N. Doinikov, A. Duke, V. Kokotkov, M. Korol'kov, V. Kukhtin, A. Panin, S. Sytchevsky, "Mechanical Forces Simulation and Stress Analysis of the TEXTOR Vacuum Vessel during Plasma Disruption under 3d Eddy Currents Load," IEEE Trans. Magnetics, Vol. 32, No. 4, July 1996.

[5] J-F. Luy and J. Schmidl, "Temperature Distribution in Cylinder Symmetric MM-Wave Devices," IEEE Trans. Microwave Theory and Techniques, Vol. 42, No. 4, April 1994, pp. 573-578.

*[6] Hughes Aircraft Company, "Reliability Assessment of Wafer Scale Integration Using Finite Element Analysis," RL-TR-91-251, Final Technical Report, October 1991.

[7] A. B. Shapiro, "TOPAZ3D–A Three-Dimensional Finite Element Heat Transfer Code," University of California, Lawrence Livermore National Laboratory Rept. UCID-20484 (1985).

[8] K. Preis, O. Biro, R. Dyczij-Edlinger, K. R. Richter, "Application of FEM to Coupled Electric, Thermal and Mechanical Problems," IEEE Trans. Magnetics, Vol. 30, No. 5, September 1994, pp. 3316-319. Also, Int. J. Applied Electromagnetics in Materials, Vol. 3, (1992), pp. 151-155.

[9] W. Kerner, "Algorithms and Software for Linear and Nonlinear MHD Simulations," Computer Physics Reports, Vol. 12 (1990), pp. 135-175.

[10] E. Nakamae, "Display of Distributed Scalars and Vectors," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 755-760.

[11] P. Girdinio, G. Molinari, and S. Ridella, "Trends in Computer Hardware and Software for Computer Aided Design Applications," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 761-766.

[12] A. Nicolas and L. Nicolas, "3D Automatic Mesh Generation for Boundary Integral Method," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 767-770.

[13] P. L. George, F. Hecht, and E. Saltel, "Automatic 3D Mesh Generation with Prescribed Meshed Boundaries," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 771-774.

[14] Y. Tanizume, H. Yamashita, and E. Nakamae, "Tetrahedral elements Generation Using Topological Mapping and Space Dividing for 3-D Magnetic Field FEM," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 775-778.

*RL TR-91-251 is Distribution limited to U.S. Government Agencies & their contractors; critical technology.

[15] T. Tärnhuvud, K. Reichert, and J. Skoczylas, "Problem-Oriented Adaptive Mesh-Generation for Accurate Finite-Element Calculation," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 779-782.

[16] D-K. Park, C-H. Ahn, S-Y. Lee, and J-W. Ra, "Adaptive Mesh Refinement for Boundary Element Method and its Application to Stripline Analysis," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 783-786.

[17] G. Henneberger, G. Meunier, J. C. Sabonnadière, "Sensitivity Analysis of the Nodal Position in the Adaptive Refinement of Finite Element Meshes," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 787-790.

[18] S. R. H. Hoole, "Eigen Value and Eigen Vector Perturbation and Adaptive Mesh Generation in the Analysis of Waveguides," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 791-794.

[19] P. Fernandes, P. Girdinio, P. Molfino, G. Molinari, M. Repetto, "A Comparison of Adaptive Strategies for Mesh Refinement Based on *a posteriori* Local Error Estimation Procedures," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 795-798.

[20] H. Tsuboi, T. Shimotsukasa, and K. Kato, "Triangular Mesh Generation Using Knowledge Base for Three-Dimensional Boundary Element Method," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 799-802.

[21] C. S. Brett, C. M. Saldanha, D. A. Lowther, "Interval Mathematics for Knowledge-Based Computer Aided Design in Magnetics," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 803-806.

[22] Y. S. Sun, G. S. Zhang, and W. Chen, "A Data Base Method for 3D Open Boundary Field Computation," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 807-810.

[23] B. T. Chia, J. L. Coulomb, J. C. Sabonnadière, J. C. Vérité, and A. Nicolas, "A Unified Database for Data Exchange in Electromagnetic Field Computation," IEEE Trans. Magnetics, Vol. 26, No. 2, March 1990, pp. 811-814.

[24] P. Fernandes, P. Girdinio, G. Molinari, and M. Repetto, "Local Error Estimation Procedures as Refinement Indicators in Adaptive Meshing," IEEE Trans. Magnetics, Vol. 27, No. 5, September 1991, pp. 4189-4192.

[25] T. W. Nehl and D. A. Field, "Adaptive Refinement of First Order Tetrahedral Meshes for Magnetostatics Using Local Delaunay Subdivisions," IEEE Trans. Magnetics, Vol. 27, No. 5, September 1991, pp. 4193-4196.

[26] K. Reichert, J. Skoczylas, and T. Tärnhuvud, "Automatic Mesh Generation Based on Expert-System-Methods," IEEE Trans. Magnetics, Vol. 27, No. 5, September 1991, pp. 4197-4200.

[27] C-H. Ahn, S-S. Lee, H-J. Lee, and S-Y. Lee, "A Self-Organizing Neural Network Approach for Automatic Mesh Generation," IEEE Trans. Magnetics, Vol. 27, No. 5, September 1991, pp. 4201-4204.

[28] N. A. Golias and T. D. Tsiboukis, "Three-Dimensional Automatic Adaptive Mesh Generation," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1700-1703.

[29] S. McFee and J. P. Webb, "Adaptive Finite Element Analysis of Microwave and Optical Devices using Hierarchal Triangles," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1708-1711.

[30] A. Kost and L. Jaenicke, "Universal Generation of an Initial Mesh for Adaptive 3-D Finite Element method," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1735-1738.

[31] P. Fernandes, P. Girdinio, M. Repetto, and G. Secondo, "Refinement Strategies in Adaptive Meshing," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1739-1742.

[32] K. G. Mitchell and J. Penman, "Self Adaptive Mesh Generation for 3-D Finite Element Calculation," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1751-1754.

[33] J. M. Zhou, K. D. Zhou, and K. R. Shao, "Automatic Generation of 3D Meshes for Complicated Solids," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1759-1762.

[34] D. N. Dyck, D. A. Lowther, and S. McFee, "Determining an Approximate Finite Element Mesh Density Using Neural Network Techniques," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1767-1770.

[35] N. Bamps, H. Hedia, A. Genon, W. Legros, A. Nicolet, and M. Ume, "Graphical Language Interpreter and Meshing Tool for Electromagnetic Computations," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1771-1773.

[36] J. F. Ballay, F. Nuwendam, and J. C. Kieny, "A Tool Using an Object Oriented Language for Field Computation in a CAD Prospect," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1774-1777.

[37] A. Stochniol, E. M. Freeman, and D. A. Lowther, "A User Oriented Shell for Electromagnetic CAD of Axisymmetric Devices," IEEE Trans. Magnetics, Vol. 28, No. 2, March 1992, pp. 1782-1784.

[38] B. K. Cabral, G. W. Laguna, R. R. McLeod, S. L. Ray, S. T. Pennock, R. L. Berger, and M. F. Bland, "Interactive Pre- and Post-Processing Tools for Finite-Difference Time-Domain Codes," IEEE AP-S/URSI Symposium, San Jose, CA, June 1989, pp. 1098-1099.

[39] K. C. Chellamuthu and Nathan Ida, "Reliability Assessment of an '*a posteriori*' Error Estimate for Adaptive Computation of Electromagnetic Field Problems," IEEE Trans. Magnetics, Vol. 31, No. 3, May 1995, pp. 1761-1764.

[40] F. J. C. Meyer and D. B. Davidson, "A Posteriori Error Estimates for Two-Dimensional Electromagnetic Field Computations: Boundary Elements and Finite Elements," Applied Computational Electromagnetics Society Journal, July 1996, Vol. 11, No. 2, pp. 40-54.

[41] J. M. Zhou, K. R. Shao, and Z. Ren, "hp-Version Adaptive Analysis of Nonlinear Magnetic Fields with Quadrilaterals," IEEE Trans. Magnetics, Vol. 31, no. 3, May 1995, pp. 1745-1748.

[42] O. C. Zienkiewicz and J. Z. Zhu, "A simple error estimator and adaptive procedure for practical engineering analysis," Int. J. Numerical Methods in Engineering, Vol. 24, pp. 337-357, 1987.

[43] N. A. Golias and T. D. Tsiboukis, "3-D Eddy-Current Computation with a Self-Adaptive Refinement Technique," IEEE Trans. Magnetics, Vol. 31, No. 3, May 1995, pp. 2261-2268.

[44] N. A. Golias and T. D. Tsiboukis, "An approach to refining three-dimensional tetrahedral meshes based on Delaunay transformations," Int. J. Num. Meth. Eng., Vol. 37, pp. 793-812, 1994.

[45] H. Kim, S. Hong, K. Choi, H. Jung, and S. Hahn, "A three dimensional adaptive finite element method for magnetostatic problems," IEEE Trans. Magnetics, Vol. 27, pp. 4193-4196, September 1991.

[46] M. Drayton, "TOPAZ2D Validation Status Report: December 1990," Lawrence Livermore National Laboratory, UCRL-ID-106550, December 1990.

[47] J. Wimp, *Sequence Transformations and Their Applications.* New York:Academic Press, 1981.

[48] S. Singh and R. Singh, "A Convergence Procedure for Computing Slowly Converging Series," IEEE Trans. Microwave Theory and Techniques, Vol. 40, No. 1, January 1992, pp. 168-171.

*U.S. GOVERNMENT PRINTING OFFICE: 1997-509-127-61002

MISSION OF ROME LABORATORY

Mission. The mission of Rome Laboratory is to advance the science and technologies of command, control, communications and intelligence and to transition them into systems to meet customer needs. To achieve this, Rome Lab:

a. Conducts vigorous research, development and test programs in all applicable technologies;

b. Transitions technology to current and future systems to improve operational capability, readiness, and supportability;

c. Provides a full range of technical support to Air Force Material Command product centers and other Air Force organizations;

d. Promotes transfer of technology to the private sector;

e. Maintains leading edge technological expertise in the areas of surveillance, communications, command and control, intelligence, reliability science, electro-magnetic technology, photonics, signal processing, and computational science.

The thrust areas of technical competence include: Surveillance, Communications, Command and Control, Intelligence, Signal Processing, Computer Science and Technology, Electromagnetic Technology, Photonics and Reliability Sciences.